

Consiglio Nazionale delle Ricerche

Un ambiente di lavoro che manipola  
files packed e procedure di utilità  
per l'ambiente XEDIT

R. Beltrame

189

**CNUCE**

A cura di : Renzo Beltrame

Copyright - Gennaio 1983

by - CNUCE - Pisa

Istituto del Consiglio Nazionale delle Ricerche

**UN AMBIENTE DI LAVORO CHE MANIPOLA FILES PACKED E PROCEDURE  
DI UTILITA' PER L'AMBIENTE XEDIT.**

Renzo BELTRAME

CNUCE - Ist. del CNR  
36, Via S. Maria  
56100 PISA - Italy

Doc. A-189

Dicembre 1982

(c) Copyright CNUCE, Pisa 1982

INDICE

1.0 Procedure di utilita' per l'ambiente Xedit. . . . . 1  
2.0 Un ambiente di lavoro che manipola files packed. . . . . 5  
Manuali di riferimento. . . . . 16

1.0 PROCEDURE DI UTILITA' PER L'AMBIENTE XEDIT.

Si illustrano alcune procedure EXEC legate al nuovo editore del CMS SP, l'XEDIT (vedi "R.2" pag. 16), che facilitano alcune operazioni di editing. Si tratta di funzioni da associare tipicamente alle PFK e adatte ad un uso "full-screen" dell'editore; esse pertanto valgono nel caso di terminali dotati di queste caratteristiche.

Due procedure si riferiscono ai comandi normalmente associati nella seduta di edit alla PF02 e alla PF12. Altre due ai comandi FILE e QUIT dell'editore.

Alla PF02 e alla PF14 e' associato abitualmente il comando SOS LINEADD. E' comodo in molti casi, soprattutto per chi usa linguaggi Algol-like e preferisce gestire in proprio l'indentazione, avere il cursore posizionato sotto l'inizio della riga precedente cosi' da mantenere l'incolonnamento. L'uso del tabulatore permette poi di gestire facilmente l'indentazione.

Per ottenere questo risultato si puo' definire una procedura di tipo XEDIT come la seguente:

```
TRANSFER CURSOR
&READ VARS &A &B &C &D
&IF &C NE -1 &GOTO -S1
EMSG LDP561E CURSOR IS NOT IN VALID DATA FIELD
&EXIT 3
-S1 SOS LINEADD
AL = &C + 1
&IF &C EQ 0 &GOTO -EE
SPAN ON BLANK *
MSGM OFF
LOCATE :&L
CL --/ /
&IF &RETCODE EQ 0 &GOTO -D1
CF
&GOTO -EE
-D1 UP
TRANSFER TRUNC
&READ VARS &A
CL :&A
-CC CL --/ /
TRANSFER LINE
&READ VARS &C
-EE SPAN OFF BLANK 2
MSGMODE ON LONG
LOCATE :&L
&IF &C EQ 0 CF
CURSOR COLUMN
```

la procedura e' abbastanza complicata per tener conto dell'incolonnamento anche quando vi sono righe bianche interposte.

Alla PF12 e alla PF24 e' associato abitualmente il comando CURSOR COLUMN. E' comodo poter disporre di una funzione un poco piu' sofisticata: che operi cio' nel modo tradizionale quando il cursore e' nel campo comandi, ma che faccia diventare la posizione del cursore sulla parte dello schermo riservata ai dati la posizione corrente del "column pointer" e mandi il cursore nel campo comandi. In questo modo si possono utilizzare meglio i comandi legati al "column pointer" - CINSERT, CDELETE, COVERLAY - molto comodi, soprattutto in fase di correzione di un testo, quando si deve intervenire a livello di caratteri singoli.

Una procedura, di tipo XEDIT, che consente la funzione descritta e' la seguente:

```
TRANSFER CURSOR
&READ VARS &A &B &C &D
&IF &C GT 0 &GOTO -P1
CURSOR COLUMN
&EXIT
-P1 &IF &D GT 0 &GOTO -T1
MSG CRL561E CURSOR IS NOT IN VALID DATA FIELD
CURSOR CMDLINE
&EXIT 3
-T1 TRANSFER CURLINE
&READ VARS &C
&C = &A - &C
&IF &C GT 0 &C = &C - 1
&C
CLOCATE :&D
CURSOR CMDLINE
```

Nelle procedure riportate si e' fatto largo uso della primitiva &GOTO poiche' una funzione, applicabile alle procedure di tipo XEDIT e a quelle di tipo EXEC elaborate dall'EXEC2, trasforma la "label" associata al GOTO nel numero di riga corrispondente e l'indirizzamento per numero di riga permette una maggior velocita' di esecuzione. Per avere questo tipo di "ottimizzazione" basta dare il comando VMFOPT prima del comando FILE; ovviamente la procedura cosi' trasformata non sopporta aggiunte, cancellazioni o spostamenti di righe. Prima di eseguire qualsiasi modifica e' opportuno dare il comando VHFDEOPT in ambiente XEDIT per ripristinare la situazione originaria.

## CHUCE A-189

La procedura precedente, dopo "ottimizzazione" si presenta così:

```

*%OPTIMIZED AT 18:17:01 ON 82/07/09
*%      N O T I C E:
*% THIS STATEMENT IS NECESSARY FOR THE OPTIMIZATION PROGRAM - VMFO
PT
*% DE-OPTIMIZE THIS MACRO BEFORE MAKING ANY CHANGES USING - VMFDEO
PT
TRANSFER CURSOR
&READ VARS &A &B &C &D
&IF &C GT 0 &GOTO 15 -F1
CURSOR COLUMN
&EXIT
-F1 &IF &D GT 0 &GOTO 19 -T1
MSG CRL561E CURSOR IS NOT IN VALID DATA FIELD
CURSOR CMDLINE
&EXIT 3
-T1 TRANSFER CURLINE
&READ VARS &C
&C = &A - &C
&IF &C GT 0 &C = &C - 1
&C
CLOCATE :&D
CURSOR CMDLINE

```

Se si associano i comandi FILE e QQUIT a delle PFK si ha l'inconveniente di una scarsa protezione contro accidentali e non voluti invii dei rispettivi comandi con effetti che possono diventare assai spiacevoli. Le due procedure che si riportano di seguito, richiedono di confermare tramite la relativa PFK il comando affinché questo diventi operativo.

Per il comando FILE la procedura - qui chiamata LFILE XEDIT - può avere la forma seguente:

```

SET PF01 FILE
SET PF13 FILE
SET RESERVED 3 H +++++ Press PF01/13 again to confirm FILE

```

dove si è supposto di assegnare il comando alla PF01 e alla PF13.

## CHUCE A-189

Per il comando QUIT la procedura - qui chiamata LQUIT XEDIT - puo' essere la seguente:

```
&IF &INDEX > 0 &GOTO -BADOPS
&R = 0
SET MSGM OFF
PQUIT
&R = &RETCODE
SET MSGM ON LONG
&IF &R = 1 &GOTO -LASTQUIT
* if the current file is the only file being edited,
* the quit command must be stacked.
&IF &R = 0 &EXIT
RESERVED 3 H +++++ File has been changed - press PF03/15 again to
QQUIT file
SET PF03 QQUIT
SET PF15 QQUIT
&EXIT
-LASTQUIT &STACK QQUIT
&EXIT
-BADOPS EMSG LQ2012S INVALID PARAMETER(S): &ARGSTRING
&EXIT 0012
```

dove si ha una struttura piu' complicata per tener conto dei casi di edit multiplo. Qui il comando e' supposto associato alla PF03 e alla PF15 su cui e' gia' definito il comando QUIT. In entrambe queste procedure si e' fatto uso del comando SET RESERVED sulla linea 3 dello schermo per evitare che messaggi di errore inviati dall'XEDIT accodandosi con il messaggio inviato dalle procedure indicate richiedano due righe e, pertanto, danno origine ad una schermata vuota di testo e contenente solo i messaggi; una situazione particolarmente scomoda nel lavoro di edit. Cio' comporta di avere definito come "default" in un "profile" un SET RESERVED 3 H che lascia disponibile la terza riga dello schermo.

Cio' e' stato fatto ad esempio nella procedura DEFAULT XEDIT impiegata dalla PROFILE XEDIT riportata nel capitolo seguente.

2.0 UN AMBIENTE DI LAVORO CHE MANIPOLA FILES PACKED.

Il nuovo editore del CMS SP, l'XEDIT, offre la possibilità di definire il formato del record - FP o VP - in modo che alla fine della sessione di edit il file venga automaticamente "impaccato" con il formato previsto dall'opzione "PACK" del comando "Copyfile". Il risparmio di spazio disco per la memorizzazione, soprattutto nel caso di files contenenti il sorgente di programmi, è ragguardevole, per cui può essere interessante avvalersi di tale possibilità per costruire un ambiente di lavoro che opera su files "packed".

Per questo si richiede che quando si usino comandi, come PRINT, TYPE, PUNCH, o i comandi che provvedono alla compilazione, il file riprenda temporaneamente la forma non impaccata.

L'ambiente proposto prevede di utilizzare una serie di procedure EXEC, che si illustrano a titolo di esempio, e che consentono di usare in modo trasparente questi comandi: senza cioè doversi preoccupare della forma del file e delle caratteristiche - R/O o R/W - del minidisco su cui risiedono.

Il nucleo di questo ambiente è costituito da tre procedure: UNPACK2 EXEC, UCMND2 EXEC e UCMND12 EXEC. La prima provvede a impaccare e disimpaccare i files ed è usata dalle altre due che invece consentono di eseguire comandi CMS secondo il duplice formato:

```
<comando> <fileid> (<opzioni>
<comando> <fn> (<opzioni>
```

La procedura UNPACK2 EXEC svolge le seguenti funzioni:

- provvede a disimpaccare, se necessario, il file richiesto; ciò avviene sullo stesso minidisco se il file è su un minidisco acceduto R/W, sul primo dei minidischi acceduti R/W se il file risiede su un minidisco acceduto R/O
- assegna al file risultato lo stesso nome o un nome ottenuto facendo precedere i caratteri \$# al nome del file originario con troncamento a otto caratteri, a seconda che il file risultato risieda sullo stesso minidisco di quello originario o no
- pone sullo stack di input fn ft fm del file risultato
- assegna al codice di ritorno un valore che permette di sapere se il file originario era o no impaccato

una sua forma possibile è la seguente:

CNUCE A-189

```

&TRACE OFF
* Performs pack/unpack of CMS files
* R. Beltrame - CNUCE
&IF &INDEX GT 1 &GOTO -IN
&TYPE Incomplete fileid specification
&EXIT 0012
-IN &PACK = 1
&IF &INDEX EQ 2 &GOTO -AA
&IF &3 NE * &GOTO -FF
&TYPE Filemode * - 1st &1 &2 fm processed
&GOTO -AA
* FN FT FM given
-FF STATE &1 &2 &3
&IF &RETCODE EQ 0 &GOTO -AAN
&TYPE File &1 &2 &3 not found
&EXIT &RETCODE
* Search for filemode
-AA &ALF = &STRING OF ABCDEFGHIJKLMNOPQRSTUVWXYZ
&J = 1
&LOOP -LL WHILE &J LE 26
&FM = &PIECE OF &ALF &J 1
STATE &1 &2 &FM
&RTCA = &RETCODE
&IF &RTCA EQ 0 &GOTO -AAC
-LL &J = &J + 1
&TYPE File &1 &2 * not found
&EXIT &RTCA
-AAC &ARGS &1 &2 &FM
-AAN &ARGS &1 &2 &3 &1 &2 &3
STATE CMS EXEC A
&IF &RETCODE EQ 0 RENAME CMS EXEC A $#CMS = =
L &1 &2 &3 (L E
* FM REFERS TO R/W DISK
FINDSTAK CMS EXEC A C38 800_
&CECK = &RETCODE
&IF &CECK NE 0 FINDSTAK CMS EXEC A C37 1024_
&CECK = &RETCODE
&IF &CECK NE 0 FINDSTAK CMS EXEC A C37 2048_
&CECK = &RETCODE
&IF &CECK NE 0 FINDSTAK CMS EXEC A C37 4096_
&CECK = &RETCODE
ERASE CMS EXEC A
STATE $#CMS EXEC A
&IF &RETCODE EQ 0 RENAME $#CMS EXEC A CMS = =
&IF &CECK NE 0 &GOTO -EE1
&READ VARS &A1 &A2 &A3 &A4 &A5 &A6 &A7 &A8 &A9 &A10
&PACK = 0
SADT &3
&RETCOD = &RETCODE
&IF &RETCOD GT 1 &EXIT &RETCOD
&READ VARS &A1 &A2 &A3 &A4 &A5 &A6 &A7 &A8 &A9 &A10
&IF &RETCOD EQ 0 &GOTO -EE

```

CHUCE A-189

```

-RO SADT ??
* FM REFERS TO R/O DISK
&IF &RETCODE EQ 0 &GOTO -RR
&READ VARS
&TYPE No R/W disk accessed
&EXIT 0012
-RR &READ VARS &A1 &A2 &A3 &A4 &A5 &A6 &A7 &A8 &A9 &A10
&6 = &CONCAT OF &A5 1
&4 = &CONCAT OF # $ &1
&ARGS &1 &2 &3 &4 &5 &6
-EE EX UNPACK &1 &2 &3 &4 &5 &6
&IF &RETCODE EQ 0 &PACK = 0
-EE1 &SA = &STRING OF &4 &5 &6
&STACK &SA
&EXIT &PACK

```

In essa si e' fatto uso delle possibilita' offerte dall'EXEC2 e dall'IPF.

La procedura UCMND2 EXEC ha:

- quale primo parametro il nome del comando CMS da eseguire
- quali ulteriori parametri il <fileid> e le eventuali opzioni nella forma consueta.

Essa ha cioe' quali parametri il comando con la sua sintassi abituale. La procedura ha la forma seguente:

```

&TRACE OFF
&CMD = &1
&ARGS &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 &13 &14
&UARGS = &1
&J = 1
&N = &INDEX - 1
&LOOP -LL &N
&J = &J + 1
&X = &LOCATION OF ( &&J
&IF &X EQ 1 &GOTO -PP
&UARGS = &STRING OF &UARGS &&J
-LL &CONTINUE
-PP &IF &J LE 4 &IF &J GT 1 &GOTO -EE
&TYPE &UARGS - Incorrect fileid specification
&EXIT 0012
-EE &IF &J NE &INDEX &GOTO -CC
&XARGS = (
&GOTO -CONT
-CC &XARGS = &RANGE OF & &J &INDEX
-CONT EX UNPACK2 &UARGS
&PAK = &RETCODE
&IF &PAK GT 1 &EXIT &PAK
&READ VARS &FN &FT &FM
STATE &CMD EXEC *

```

CNUCE A-189

```
&IF &RETCODE EQ 0 &CMD = &STRING OF EX &CMD
&CMD &FN &FT &PM &XARGS
&RETCOD = &RETCODE
&IF &FN EQ &1 &GOTO -NN
ERASE &FN &FT &PM
CLRSCRN
&EXIT &RETCOD
-NN &IF &PAK EQ 0 EX PACK &FN &FT &PM
CLRSCRN
&EXIT &RETCOD
```

Anche qui si e' fatto uso delle possibilita' offerte dall'EXEC2. Eseguito il comando la procedura provvede a impaccare di nuovo il file originario quando necessario.

Mediante questa procedura e' possibile definire facilmente i comandi TYPE, PRINT, PUNCH, etc. La procedura UTYPE EXEC che ridefinisce il comando TYPE ha la forma seguente:

```
&CONTROL OFF NOMSG
EX UCMND2 TYPE &1 &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 &13
&EXIT &RETCODE
```

La procedura UPRINT EXEC che ridefinisce il comando PRINT sara':

```
&CONTROL OFF NOMSG
EX UCMND2 PRINT &1 &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 &13
&EXIT &RETCODE
```

e analogamente per altre similari.

Per il caso di comandi che prevedono tra i parametri il solo fn, come ad esempio i comandi che richiamano i compilatori dei vari linguaggi di programmazione, si e' prevista la procedura UCMND12 EXEC, di cui una forma possibile e' la seguente:

```
&TRACE OFF
* Performs CMS command (Fn only)
* with packed/unpacked files
&CMD = &1
&ARGS &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12 &13 &14
&UARGS = &1
&J = 1
&N = &INDEX - 1
&LOOP -LL &N
&J = &J + 1
&X = &LOCATION OF ( &&J
&IF &X EQ 1 &GOTO -PP
&UARGS = &STRING OF &UARGS &&J
-LL &CONTINUE
-PP &IF &J LE 4 &GOTO -EE
&TYPE &UARGS - Incorrect fileid specification
```

```

&EXIT 0012
-EE &IF &J GT 2 &GOTO -EE1
&IF &CMD EQ SINCHS &PT = SINULA
&IF &CMD EQ PLIOPT &PT = PLI
&IF &CMD EQ PORTGI &PT = PORTRAN
&IF &CMD EQ FORTHX &PT = FORTRAN
&IF &CMD EQ ASH &PT = ASSEMBLE
&IF &CMD EQ SCRIPT &PT = SCRIPT
&IF &CMD EQ IBMS &PT = SCRIPT
&IF &CMD EQ PASCALVS &PT = PASCAL
&IF &CMD EQ PAS8000 &PT = PASCAL
&IF &CMD EQ A68C &PT = ALGOL68C
* The list is open to addenda
&UARGS = &STRING OF &1 &PT
STATE &1 &PT *
&IF &RETCODE EQ 0 &GOTO -EE1
&TYPE &UARGS not found
&EXIT &RETCODE
-EE1 &IF &J NE &INDEX &GOTO -CC
&KARGS = (
&GOTO -CONT
-CC &KARGS = &RANGE OF & &J &INDEX
-CONT &CONTINUE
&IF .&PT NE . ERASE &1 TEXT *
EX UNPACK2 &UARGS
&PAK = &RETCODE
&IF &PAK GT 1 &EXIT &PAK
&READ VARS &FN &PT &FM
STATE &CMD EXEC *
&IF &RETCODE EQ 0 &CMD = &STRING OF EX &CMD
&CMD &FN &KARGS
&IF &RETCODE LT 0 &RETCODE = 16
&IF &RETCODE GE 16 &RETCODE = 16
&RETC D = &RETCODE
&IF &FN EQ &1 &GOTO -NN
ERASE &FN &PT &FM
CLRSCRN
&EXIT &RETC D
-NN &IF &PAK EQ 0 EX PACK &FN &PT &FM
CLRSCRN
&EXIT &RETC D

```

Un esempio di procedura EXEC per attivare i compilatori puo' essere la seguente, dove vengono anche stabilite alcune opzioni che l'utente desidera costantemente attive:

```

&TRACE OFF
&CMD = &1
&UARGS = &2
&J = 2
&N = &INDEX - 1
&LOOP -LL &N

```

CNUCE A-189

```

&J = &J + 1
&X = &LOCATION OF ( &&J
&IF &X EQ 1 &GOTO -PP
&UARGS = &STRING OF &UARGS &&J
-LL &CONTINUE
-PP &IF &J LE 5 &GOTO -EE
&TYPE &UARGS - Incorrect fileid specification
&EXIT 0012
-EE &IF &J GE &INDEX &GOTO -CONT
&XARGS = &RANGE OF & &J &INDEX
&XARGS = &SUBSTR OF &XARGS 2 *
-CONT &CONTINUE
&IF &COMD EQ SIMCMS &LARGS = &STRING OF (RESWD 3 XREF NOGO PRINT &
XARGS
&IF &COMD EQ PASCALVS &LARGS = &STRING OF (PXREF XREF PRINT &XARGS
&IF &COMD EQ PAS8000 &LARGS = &STRING OF (PRINT OBJ &XARGS
&IF &COMD EQ A68C &LARGS = &STRING OF (PRINT XREF &XARGS
EX UCMND12 &COMD &UARGS &LARGS
&EXIT &RETCODE

```

Se si desidera ridefinire i comandi standard che richiamano i compilatori basta definire procedure del tipo:

```

&CONTROL OFF NOMSG
EX UCOMP2 ASSEMBLE &1 &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12
&EXIT &RETCODE

```

```

&CONTROL OFF NOMSG
GLOBPOP
GL TXTLIB PASCALVS CMSLIB
EX UCOMP2 PASCALVS &1 &2 &3 &4 &5 &6 &7 &8 &9 &10 &11 &12
&RETC = &RETCODE
GLOBPUSH
&EXIT &RETC

```

Per impiegare l'ambiente qui descritto vanno modificate le opzioni standard dell'editore legate ai vari ft. Qui di seguito si offre un esempio di Profile molto generale, essa consente infatti di ridefinire ogni tipo di opzione, evitando anche alcune limitazioni che si avrebbero operando direttamente con la Profile.

```

&IF &INDEX < 2 &EXIT 0008
&FM = *
&IF &INDEX = 2 &GOTO -DF
&X = &POSITION OF ( &3
&IF &X = 0 &GOTO -DF
&FM = &3
-DF &STACK DEFAULT
&IF &2 NE SIMULA &IF &2 NE PASCAL &GOTO -XL
&STACK DEFSIM

```

Un ambiente di lavoro che manipola files packed.

CHUCE A-189

```
LOAD &1 &2 &FM
&EXIT &RETCODE
-XL &IF &2 NE EXEC &IF &2 NE XEDIT &GOTO -PL
&STACK DEPEX
LOAD &1 &2 &FM
&EXIT &RETCODE
-PL &IF &2 NE PLI &IF &2 NE PLIOPT &GOTO -MS
&STACK DEFSIM
LOAD &1 &2 &FM
&EXIT &RETCODE
-MS &IF &2 NE MEMO &IF &2 NE SCRIPTU &GOTO -LS
&STACK DEFSIM
LOAD &1 &2 &FM
&EXIT &RETCODE
-LS &IF &2 NE LISTING &GOTO -SC
&STACK DEPLIST
LOAD &1 &2 &FM
&EXIT &RETCODE
-SC &IF &2 NE SCRIPT &GOTO -AF
&STACK DEFSCRI
LOAD &1 &2 &FM
&EXIT &RETCODE
-AF &IF &2 NE ASSEMBLE &IF &2 NE PORTRAN &GOTO -SR
&STACK DEFPACK
LOAD &1 &2 &FM
&EXIT &RETCODE
-SR &IF &1 NE JOB &GOTO -ST
&STACK DEFJOB
LOAD &1 &2 &FM
&EXIT &RETCODE
-ST &IF &2 NE UVS &IF &2 NE JOB &IF &2 NE MVS &GOTO -LD
&STACK DEFMVS
LOAD &1 &2 &FM
&EXIT &RETCODE
-LD &IF &2 NE ALGOL68C &IF &2 NE ALGOL68 &GOTO -FIN
&STACK DEFA68C
LOAD &1 &2 &FM
&EXIT &RETCODE
-FIN LOAD &1 &2 &FM
&EXIT &RETCODE
```

A titolo di esempio riportiamo qui la DEFPACK XEDIT:

```
RECFM FP
CASE U
LRECL 80
SPAN ON BLANK 2
ZONE 1 72
TRUNC 72
SERIAL ALL 100 100
```

e la DEFSIM XEDIT:

CHUCE 1-189

```
RECFM FP
LRECL 80
ZONE 1 72
TRUNC 72
LINEND OFF
TABS 1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 6
4 67 72
SERIAL ALL 100 100
```

Vi e' poi una DEFAULT XEDIT nella quale sono inserite una serie di opzioni che personalizzano l'ambiente Xedit tenendo conto dell'attivita' prevalente, del tipo di terminale usato, etc. Nel caso personale considerato tale procedura ha la forma seguente:

```
CMS CLRSCRN
STAY ON
NONDISP
SYN / 1 CLOCATE/
SYN POWERINP 3 POWERINP
SYN XEDIT 5 XEDIT
IMP ON
ARBCHAR ON $
LINEND ON #
AUTOSAVE 5
RESERVED 3 H
V 1 72
SERIAL OFF
CASE M
PF01 LFILE
PF02 LADDPOS
PF03 LQUIT
PF06 SOS TABB 1
PF07 UP 16
PF08 NEXT 16
PF10 SPLIT CURSOR
PF11 JOIN CURSOR
PF12 CURLOC
PF13 LFILE
PF14 LADDPOS
PF15 LQUIT
PF17 SCHANGE 18
PF18 SOS TABB 1
PF19 UP 16
PF20 NEXT 16
PF22 SPLIT CURSOR
PF23 JOIN CURSOR
PF24 CURLOC
```

Per utilizzare in modo trasparente i vari comandi e' possibile ricorrere alla loro ridefinizione mediante il comando SYNONYM. Si

CHUCE A-189

da' qui un esempio di una procedura EXEC che permette di ottenere questo effetto.

```
&CONTROL OFF NOMSG
&CONTINUE
-PAN &TYPE Environment? (0/1/2 normal/pack/update):
&READ VARS &IN
&IF .&IN EQ . &GOTO -PAN
&IF &IN EQ 0 &GOTO -NORM
&IF &IN EQ 1 &GOTO -NPACK
&IF &IN EQ 2 &GOTO -UPDL
&GOTO -PAN
-NORM SYN MYSYNON SYNONYM * (STD CLEAR
CLRSCRN
&EXIT 0
-NPACK SYN USYNON SYNONYM * (NOSTD CLEAR
&EXIT 0
-UPDL SYN UPSYNON SYNONYM * (NOSTD CLEAR
CLRSCRN
&EXIT 0
```

dove e' previsto l'aggancio anche ad un terzo ambiente che sfrutta le possibilita' di produzione automatica di files di update dell'xedit.

Il file USYNON SYNONYM puo' avere la forma seguente:

ACCESS	ACCESS	2
ACD	ACD	3
AMSERV	AMSERV	2
ASSEMBLE	ASSEMBLE	2
BASIC	BASIC	3
COBOL	COBOL	2
COMPARE	COMPARE	3
COPYFILE	COPYFILE	4
DIRMAINT	DIRMAINT	4
DOSLKED	DOSLKED	4
EDGAR	EDGAR	3
EDIT	EDIT	1
ENVRNMT	ENVRNMT	3
EXEC	EXEC	2
FETCH	FETCH	3
FILEDEF	FILEDEF	2
FLIST	FLIST	2
GENMOD	GENMOD	1
GLOBAL	GLOBAL	2
HELP	HELP	1
INCLUDE	INCLUDE	2
LABELDEF	LABELDEF	2
LISTFILE	LISTFILE	1
LOADMOD	LOADMOD	5
MACLIB	MACLIB	3

CHUCE A-189

MODMAP	MODMAP	3
MOVEFILE	MOVEFILE	4
OPTION	OPTION	2
PFNKEY	PFNKEY	3
PLIOPT	PLIOPT	4
QUERY	QUERY	1
READCARD	READCARD	4
RELEASE	RELEASE	3
RENAME	RENAME	1
SCRIPT	SCRIPT	2
SVCTRACE	SVCTRACE	3
SYNONYM	SYNONYM	3
TXTLIB	TXTLIB	3
USSEM	ASSEMBLE	1
UA68C	ALC	3
UBROWSE	BROWU	2
UPASCVS	VPASC	3
UPDATE	UPDATE	1
UPRINT	PRIU	2
UPUNCH	PUNU	2
UPASCVS	PASVS	4
UPAS8000	PAS80	4
USIMC	SIMCU	4
UTYPE	TYPEU	1
UVS	UVS	2
VSBASIC	VSBASIC	3
XUD	XUD	1
XEDIT	XEDIT	2
XNED	XNED	2
XPED	XPED	2
ZED	ZED	2

La situazione dei comandi diventa allora la seguente:

SYSTEM COMMAND	USER SYNONYM	SHORTEST FORM (IF ANY)
-----	-----	-----
ACCESS	ACCESS	AC
ACD	ACD	
ANSERV	ANSERV	AM
ASSEMBLE	ASSEMBLE	AS
BASIC	BASIC	BAS
COBOL	COBOL	CO
COMPARE	COMPARE	COM
COPYFILE	COPYFILE	COPY
DIRMAINT	DIRMAINT	DIRM
DOSLKED	DOSLKED	DOSL
EDGAR	EDGAR	EDG
EDIT	EDIT	E
ENVRNMT	ENVRNMT	ENV
EXEC	EXEC	EX
FETCH	FETCH	FET

CHUCE 1-189

FILEDEF	FILEDEF	FI
FLIST	FLIST	PL
GENMOD	GENMOD	G
GLOBAL	GLOBAL	GL
HELP	HELP	H
INCLUDE	INCLUDE	IN
LABELDEF	LABELDEF	LA
LISTFILE	LISTFILE	L
LOADMOD	LOADMOD	LOADM
MACLIB	MACLIB	MAC
MODMAP	MODMAP	MOD
MOVEFILE	MOVEFILE	MOVE
OPTION	OPTION	OP
PFNKEY	PFNKEY	PFN
PLIOPT	PLIOPT	PLIO
QUERY	QUERY	Q
READCARD	READCARD	READ
RELEASE	RELEASE	REL
RENAME	RENAME	R
SCRIPT	SCRIPT	SC
SVCTRACE	SVCTRACE	SVC
SYNONYM	SYNONYM	SYN
TXTLIB	TXTLIB	TKT
UASSEM	ASSEMBLE	A
UA68C	ALC	
UBROWSE	BROWU	BR
UPASCVS	VPASC	VPA
UPDATE	UPDATE	U
UPRINT	PRIU	PR
UPUNCH	PUNU	PU
UPASCVS	PASVS	PASV
UPAS8000	PAS80	PAS8
USIMC	SIMCU	SIMC
UTYPE	TYPEU	T
UVS	UVS	UV
VSBASIC	VSBASIC	VSB
XUD	XUD	X
XEDIT	XEDIT	XE

Una situazione verificata dando il comando SYN in ambiente CMS. Come si puo' vedere le procedure EXEC che manipolano i files impaccati hanno le abbreviazioni tipiche dei relativi comandi CMS in ambiente normale, garantendo cosi' la trasparenza a livello utente.

I comandi tradizionali, ove necessario, sono rimasti, come si puo' vedere, con diversa abbreviazione.