File created: 5-Jun-2022 00:14:07 **{DSK}<users>kaplan>local>medley3.5>working-medley>sources>COREIO.;17**

changes to: (FNS \CORE.OPENFILE)

previous date: 4-Jun-2022 16:30:20 {DSK}<users>kaplan>local>medley3.5>working-medley>sources>COREIO.;16

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

```
;;
;; Copyright (c) 1981-1988, 1990, 1993, 1999, 2018 by Venue & Xerox Corporation.

(RPAQQ COREIOCOMS
       (

;;; Implementation of Core resident 'files'

        (FNS \CORE.CLOSEFILE \CORE.DELETEFILE \CORE.DIRECTORYNAMEP \CORE.FINDPAGE \CORE.GENERATEFILES
             \CORE.NEXTFILEFN \CORE.FILEINFOFN \CORE.GETFILEHANDLE \CORE.GETFILEINFO
             \CORE.GETFILEINFO.FROM.INFOBLOCK \CORE.GETFILENAME \CORE.GETINFOBLOCK \CORE.NAMESCAN
             \CORE.NAMESEGMENT \CORE.OPENFILE \COREFILE.SETPARAMETERS \CORE.PACKFILENAME \CORE.RELEASEPAGES
             \CORE.SETFILEPTR \CORE.UPDATEOF \CORE.BACKFILEPTR \CORE.SETEOFPTR \CORE.SETACCESSTIME
             \CORE.SETFILEINFO \CORE.GETNEXTBUFFER \CORE.UNPACKFILENAME)
        (FNS COREDEVICE \CREATECOREDEVICE)
        (FNS \NODIRCOREFDEV \NODIRCORE.OPENFILE)
        (DECLARE%: DONTCOPY (RECORDS CORE.PAGEENTRY COREFILEINFOBLK CORESTREAM COREDEVICE COREGENFILESTATE))
        (INITRECORDS COREFILEINFOBLK)
        (DECLARE%: DONTEVAL@LOAD DOCOPY (P (COREDEVICE 'NODIRCORE T)
                                           (COREDEVICE 'CORE)
                                           (COREDEVICE 'SCRATCH T)))
        (DECLARE%: DOEVAL@LOAD DONTCOPY (LOCALVARS . T))))

;;; Implementation of Core resident 'files'

(DEFINEQ

(\CORE.CLOSEFILE
  [LAMBDA (STREAM)                                                    (* hdj "22-Sep-86 18:40")

;;; Close a CORE file.

    (SELECTQ (fetch ACCESS of STREAM)
        ((OUTPUT BOTH APPEND)
            (\CORE.UPDATEOF STREAM)
            (replace IOEPAGE of (fetch INFOBLK of STREAM) with (fetch EPAGE of STREAM))
            (replace IOEOFFSET of (fetch INFOBLK of STREAM) with (fetch EOFFSET of STREAM))
            (\CORE.RELEASEPAGES STREAM (fetch EPAGE of STREAM)))
        NIL)
    (UNINTERRUPTABLY
        (replace CBUFPTR of STREAM with NIL)
        (replace CBUFSIZE of STREAM with 0))
    STREAM])

(\CORE.DELETEFILE
  [LAMBDA (FILENAME DEV EVENIFOPEN)                                   ; Edited 23-Oct-87 16:36 by bvm:
                                                                      ; delete a file from a directory.
    (PROG [(INFOBLOCK (COND
                          ((type? STREAM FILENAME)                    ; If ACCESS, it's open.
                           (AND (OR EVENIFOPEN (NULL (fetch ACCESS of FILENAME)))
                                (fetch INFOBLK of FILENAME)))
                          (T (\CORE.GETINFOBLOCK FILENAME 'OLDEST DEV]
          (COND
             ((OR (NULL INFOBLOCK)
                  (FDEVOP 'OPENP DEV (fetch IOFILEFULLNAME of INFOBLOCK)
                          NIL DEV))                                   ; Can't delete an open file
              (RETURN)))
          [for NAMETAIL on (fetch COREDIRECTORY of DEV)
             when [for EXTTAIL on (CADR NAMETAIL) when [for VERSTAIL on (CADR EXTTAIL)
                                                          when (EQ (CDR (CADR VERSTAIL))
                                                                   INFOBLOCK)
                                                          do (RETURN (RPLACD VERSTAIL (CDDR VERSTAIL]
                       do (RETURN (OR (CDADR EXTTAIL)
                                      (RPLACD EXTTAIL (CDDR EXTTAIL]
             do (RETURN (OR (CDADR NAMETAIL)
                            (RPLACD NAMETAIL (CDDR NAMETAIL]          ; Ad hoc code to Delete directory entry
          (replace IOFILEPAGES of INFOBLOCK with (LIST (create CORE.PAGEENTRY
                                                               PAGENUMBER _ 0)))
          (RETURN (fetch IOFILEFULLNAME of INFOBLOCK])
```

(\CORE.DIRECTORYNAMEP

```
[LAMBDA (DIRNAME DEV)                                          ; Edited 18-Jan-2022 11:17 by rmk
                                                               ; Edited 10-Jan-2022 22:33 by rmk
```

;; Edited 9-Jan-2022 12:42 by rmk: Using the new FILEDIRCASEARRAY so that slashes and brackets match

;; Edited 5-Jan-2022 15:03 by rmk: The previous definition didn't actually check to see if the directory existed. "existed" for COREIO means there ;; is at least one file currently in that directory.

;; Edited 19-Feb-93 16:04 by jds

```
(CL:WHEN DIRNAME
```

;; The DIRNAME could be just {CORE}, which always is OK, or {CORE}xxx. If the latter, then we want it to be a directory and not a file ;; (assuming that xxx and xxx> can't both exist.

```
    (IF (EQ (CHARCODE })
            (NTHCHARCODE DIRNAME -1))
      ELSE (CL:UNLESS (MEMB (NTHCHARCODE DIRNAME -1)
                            (CHARCODE (> /)))
                 (SETQ DIRNAME (CONCAT DIRNAME ">")))
```

;; DIRPOS because caller may not have stripped off the device. This will match the first < or / (or >)

```
    (FOR ENTRY (DIRPOS _ (STRPOS "<" DIRNAME 1 NIL NIL NIL FILEDIRCASEARRAY))
       FIRST (CL:UNLESS (EQ DIRPOS 1)
                        (SETQ DIRNAME (SUBSTRING DIRNAME DIRPOS)))
       IN (CDR (FETCH COREDIRECTORY OF DEV)) WHEN (STRPOS DIRNAME (CAR ENTRY)
                                                                  1 NIL T NIL FILEDIRCASEARRAY)
       DO (RETURN T)))])
```

(\CORE.FINDPAGE

```
[LAMBDA (STREAM PN)                                            (* bvm%: "20-Apr-85 13:32")
                                                               ; Finds the entry for page PN in the page list for STREAM,
                                                               ; creating it if necessary.
  (PROG ((CACHE (fetch COREPAGECACHE of STREAM))
         PAGETAIL PREVTAIL PAGEPTR PE)
        [SETQ PAGETAIL (COND
                         ((AND CACHE (LEQ (fetch PAGENUMBER of (CAR CACHE))
                                          PN))
```

;; Use cache: PN must be somewhere in this tail of the page list, so no sense in searching the entire page list

```
                          CACHE)
                         (T (COND
                               ((LESSP PN 0)                   ; Consistency check so that we don't try to RPLACD NIL down
                                                               ; below
                                (\ILLEGAL.ARG PN)))
                            (fetch FILEPAGES of STREAM]
    LP                                                         ; Page 0 always exists
        (COND
           [(EQ (fetch PAGENUMBER of (SETQ PE (CAR PAGETAIL)))
                PN)
            (OR (SETQ PAGEPTR (fetch PAGEPOINTER of PE))
                (replace PAGEPOINTER of PE with (SETQ PAGEPTR (\ALLOCBLOCK (FOLDHI WORDSPERPAGE WORDSPERCELL]
           [[OR (IGREATERP (fetch PAGENUMBER of PE)
                           PN)
                (NULL (SETQ PAGETAIL (CDR (SETQ PREVTAIL PAGETAIL]
```

;; PN would be before this, so it doesn't exist yet; splice it onto front of tail. This case also works when we hit the end of the list, in ;; which case we are just smashing a new cons onto the end

```
            (RPLACD PREVTAIL (SETQ PAGETAIL (CONS [create CORE.PAGEENTRY
                                                          PAGENUMBER _ PN
                                                          PAGEPOINTER _ (SETQ PAGEPTR (\ALLOCBLOCK (FOLDHI

                                                                                                   WORDSPERPAGE

                                                                                                   WORDSPERCELL
                                                                                                    ]
                                                  PAGETAIL]
           (T (GO LP)))
        (replace COREPAGECACHE of STREAM with PAGETAIL)
        (RETURN PAGEPTR])
```

(\CORE.GENERATEFILES

```
[LAMBDA (FDEV PATTERN DESIREDPROPS OPTIONS)                    (* bvm%: " 9-Jul-84 14:11")
  (PROG ((FILTER (DIRECTORY.MATCH.SETUP PATTERN))
         (DESIREDVERSION (FILENAMEFIELD PATTERN 'VERSION))
         MATCHINGFILES)
        [SETQ MATCHINGFILES (for NAME in (CDR (fetch (FDEV DEVICEINFO) of FDEV))
                               join (for EXT in (CDR NAME) when (CDR EXT)
                                       join (COND
                                               ((FIXP DESIREDVERSION)
                                                (AND (SETQ EXT (ASSOC DESIREDVERSION (CDR EXT)))
                                                     [DIRECTORY.MATCH FILTER (fetch (COREFILEINFOBLK
                                                                                      IOFILEFULLNAME)
                                                                                of (SETQ EXT (CDR EXT]
                                                     (LIST EXT)))
                                               ((DIRECTORY.MATCH FILTER (CONCAT (CAR NAME)
                                                                                ".")
```

```
                                                                    (CAR EXT)))
                                                (COND
                                                   [(NULL DESIREDVERSION)
                                                                    ; Highest version only
                                                    (LIST (CDR (CADR EXT]
                                                   (T (for VERS in (CDR EXT) collect (CDR VERS]
        (RETURN (create FILEGENOBJ
                        NEXTFILEFN _ (FUNCTION \CORE.NEXTFILEFN)
                        FILEINFOFN _ (FUNCTION \CORE.FILEINFOFN)
                        GENFILESTATE _ (create COREGENFILESTATE
                                               COREFILELST _ (CONS NIL MATCHINGFILES])
```

(\CORE.NEXTFILEFN

```
[LAMBDA (GENFILESTATE NAMEONLY SCRATCHLIST)                    (* bvm%: " 9-Jul-84 14:05")
  (PROG (FILE)
        (pop (fetch COREFILELST of GENFILESTATE))
        [SETQ FILE (fetch (COREFILEINFOBLK IOFILEFULLNAME) of (CAR (OR (fetch COREFILELST of GENFILESTATE)
                                                                       (RETURN]
        (RETURN (COND
                   (NAMEONLY (NAMEFIELD FILE T))
                   (T FILE])
```

(\CORE.FILEINFOFN

```
[LAMBDA (GENFILESTATE ATTRIBUTE)                               (* bvm%: " 3-May-84 10:50")
  (\CORE.GETFILEINFO.FROM.INFOBLOCK (CAR (fetch COREFILELST of GENFILESTATE))
         ATTRIBUTE])
```

(\CORE.GETFILEHANDLE

```
[LAMBDA (NAME RECOG FD ACCESS)                                 ; Edited 23-Oct-87 17:35 by bvm:
  (LET [(INFOBLOCK (\CORE.GETINFOBLOCK NAME RECOG FD (AND (NEQ ACCESS 'INPUT)
                                                          (SELECTQ RECOG
                                                              ((NEW OLD/NEW)
                                                               ; Yes, create it if it doesn't already exist.
                                                                    T)
                                                               NIL]
       (if INFOBLOCK
           then (if (\FILE-CONFLICT (fetch IOFILEFULLNAME of INFOBLOCK)
                                    ACCESS FD)
                    then (CL:ERROR 'XCL:FILE-WONT-OPEN :PATHNAME (fetch IOFILEFULLNAME of INFOBLOCK)))
                [if (EQ ACCESS 'OUTPUT)
                    then                                       ; Open for OUTPUT implies no content, so smash any existing
                                                               ; pages
                         (replace IOEOFFSET of INFOBLOCK with 0)
                         (replace IOEPAGE of INFOBLOCK with 0)
                         (replace IOFILEPAGES of INFOBLOCK with (LIST (create CORE.PAGEENTRY
                                                                              PAGENUMBER _ 0]
                (create CORESTREAM
                        DEVICE _ FD
                        INFOBLK _ INFOBLOCK
                        FULLFILENAME _ (fetch IOFILEFULLNAME of INFOBLOCK)
                        EOFFSET _ (fetch IOEOFFSET of INFOBLOCK)
                        EPAGE _ (fetch IOEPAGE of INFOBLOCK)
                        EOLCONVENTION _ (fetch COREEOLC of INFOBLOCK)
                        CBUFMAXSIZE _ BYTESPERPAGE])
```

(\CORE.GETFILEINFO

```
[LAMBDA (STREAM ATTRIBUTE DEV)                                 ; Edited  4-Oct-2018 14:06 by rmk:
                                                               ; Edited  9-Nov-87 14:23 by sye
```

;; Get the value of the ATTRIBUTE for a Core file. If STREAM is a filename, then the file is not open.

;; RMK: Changed so that EOL of an open stream reverts to the generic.

```
  (if (AND (type? STREAM STREAM)
           (OPENED STREAM)
           (SELECTQ ATTRIBUTE
                ((LENGTH SIZE)
                     T)
                NIL))
      then                                                     ; Let generic GETFILEINFO get this from the stream
           NIL
    else (\CORE.GETFILEINFO.FROM.INFOBLOCK (\CORE.GETINFOBLOCK STREAM 'OLD DEV)
                ATTRIBUTE])
```

(\CORE.GETFILEINFO.FROM.INFOBLOCK

```
[LAMBDA (INFOBLOCK ATTRIBUTE)                                  (* bvm%: "15-Jan-85 17:39")
  (COND
     (INFOBLOCK (SELECTQ ATTRIBUTE
                    (LENGTH (create BYTEPTR
                                    PAGE _ (fetch IOEPAGE of INFOBLOCK)
                                    OFFSET _ (fetch IOEOFFSET of INFOBLOCK)))
                    (SIZE (IPLUS (fetch IOEPAGE of INFOBLOCK)
                                 (FOLDHI (fetch IOEOFFSET of INFOBLOCK)
                                         BYTESPERPAGE)))
```

```
                    (BYTESIZE 8)
                    (CREATIONDATE (GDATE (fetch IOFIBCreationTime of INFOBLOCK)))
                    (READDATE (GDATE (fetch IOFIBReadTime of INFOBLOCK)))
                    (WRITEDATE (GDATE (fetch IOFIBWriteTime of INFOBLOCK)))
                    (ICREATIONDATE
                         (fetch IOFIBCreationTime of INFOBLOCK))
                    (IREADDATE (fetch IOFIBReadTime of INFOBLOCK))
                    (IWRITEDATE (fetch IOFIBWriteTime of INFOBLOCK))
                    ((TYPE FILETYPE)
                         (fetch IOFIBType of INFOBLOCK))
                    (EOL (SELECTC (fetch COREEOLC of INFOBLOCK)
                              (CR.EOLC 'CR)
                              (LF.EOLC 'LF)
                              (CRLF.EOLC 'CRLF)
                              (SHOULDNT)))
                    NIL])
```

(\CORE.GETFILENAME

```
[LAMBDA (NAME RECOG FD)                                   ; Edited 23-Oct-87 17:24 by bvm:
  (LET (ROOT EXT VERS SCR CREATEFLG)
       (DECLARE (SPECVARS ROOT EXT VERS))
       (\CORE.UNPACKFILENAME NAME)                        ; Sets ROOT EXT and VERS freely
       (if [AND [SETQ ROOT (CAR (OR (SETQ SCR (\CORE.NAMESCAN ROOT (fetch COREDIRECTORY of FD)
                                                  CREATEFLG))
                                    (\CORE.NAMESEGMENT ROOT]
                [SETQ EXT (CAR (OR (SETQ SCR (\CORE.NAMESCAN EXT SCR CREATEFLG))
                                   (\CORE.NAMESEGMENT EXT]
                (COND
                   [VERS                                  ; Explicit version given--must be found, or RECOG must permit
                                                          ; new file.
                         (OR (FASSOC VERS (CDR SCR))
                             (EQ RECOG 'OLD/NEW)
                             (EQ RECOG 'NEW]
                   (T                                     ; Default the version per RECOG
                      (SETQ SCR (CDR SCR))                ; Current versions, highest first.  Each element is in the form (n .
                                                          ; infoblock)
                      (SETQ VERS (SELECTQ RECOG
                                          (NEW            ; One higher than current highest
                                               (ADD1 (OR (CAAR SCR)
                                                         0)))
                                          (OLD (CAAR SCR))
                                          (OLDEST (CAAR (FLAST SCR)))
                                          (OLD/NEW        ; Highest existing version, if any, else 1.
                                                   (OR (CAAR SCR)
                                                       1))
                                          (SHOULDNT]
            then (\CORE.PACKFILENAME FD])
```

(\CORE.GETINFOBLOCK

```
[LAMBDA (NAME RECOG FD CREATEFLG)                         (* rmk%: " 5-NOV-83 21:05")
  (COND
     ((type? STREAM NAME)
      (fetch INFOBLK of NAME))
     (T (PROG (ROOT EXT VERS SCR INFOBLOCK NEWSTREAM)
              (DECLARE (SPECVARS ROOT EXT VERS))
              (\CORE.UNPACKFILENAME NAME)               ; Sets ROOT EXT and VERS freely
              (COND
                 ((SETQ SCR (\CORE.NAMESCAN ROOT (fetch COREDIRECTORY of FD)
                                    CREATEFLG))
                  (SETQ ROOT (CAR SCR))                 ; In case name completion occurred

                  )
                 (T (RETURN)))
              (COND
                 ((SETQ SCR (\CORE.NAMESCAN EXT SCR CREATEFLG))
                  (SETQ EXT (CAR SCR)))
                 (T (RETURN)))
              [COND
                 [VERS (COND
                           [(SETQ INFOBLOCK (CDR (FASSOC VERS (CDR SCR]
                           (CREATEFLG (SETQ INFOBLOCK (create COREFILEINFOBLK
                                                              IOFILEFULLNAME _ (\CORE.PACKFILENAME FD)))
                                      (for I on SCR when (OR (NOT (CDR I))
                                                             (IGREATERP VERS (CAADR I)))
                                         do (push (CDR I)
                                                  (CONS VERS INFOBLOCK))
                                            (RETURN]
                 (T (SELECTQ (COND
                                ((NEQ RECOG 'OLD/NEW)
                                 RECOG)
                                ((CDR SCR)
                                 'OLD)
                                (T 'NEW))
                             (NEW (SETQ VERS (ADD1 (OR (CAAR (CDR SCR))
                                                       0)))
```

```
                                (SETQ INFOBLOCK (create COREFILEINFOBLK
                                                        IOFILEFULLNAME _ (\CORE.PACKFILENAME FD)))
                                (push (CDR SCR)
                                      (CONS VERS INFOBLOCK)))
                           (OLD (SETQ INFOBLOCK (CDADR SCR)))
                           (OLDEST (SETQ INFOBLOCK (CDAR (FLAST SCR))))
                           (SHOULDNT]
                  (RETURN INFOBLOCK])
```

(\CORE.NAMESCAN

```
[LAMBDA (NAME NAMELST CREATEFLG)
```

;; Edited 11-Jan-2022 09:30 by rmk: Matching with FILEDIRCASEARRAY, for /

;; Edited 23-Oct-87 17:11 by bvm:

```
(COND
   ((LISTP NAMELST)
    (bind NEWSEG NEXTNAME while [AND (CDR NAMELST)
                                     (COND
                                        ((STRING.EQUAL (SETQ NEXTNAME (CAAR (CDR NAMELST)))
                                                NAME FILEDIRCASEARRAY)
                                                                    ; Found it
                                         (RETURN (CADR NAMELST)))
                                        (T (ALPHORDER NEXTNAME NAME FILEDIRCASEARRAY]
         do                                                         ; Segments are in order, so stop when (CDR NAMELST) is
                                                                    ; lexicographically greater than NAME
      (SETQ NAMELST (CDR NAMELST))
         finally (RETURN (COND
                         ((AND CREATEFLG (SETQ NEWSEG (\CORE.NAMESEGMENT NAME)))
                          (RPLACD NAMELST (CONS NEWSEG (CDR NAMELST)))
                          NEWSEG])
```

(\CORE.NAMESEGMENT

```
[LAMBDA (NAME)                                                      (* rmk%: "24-FEB-84 21:14")
```

;; Checks that name is a valid name fragment and makes a list of it if so

;; Possibly we should check the validity of each character of NAME, but for the time being we just upper case it to merge together files spelt with
;; different case letters.

```
(AND (NLISTP NAME)
     (LIST NAME])
```

(\CORE.OPENFILE

```
[LAMBDA (NAME ACCESS RECOG PARAMETERS FDEV OLDSTREAM)               ; Edited  5-Jun-2022 00:14 by rmk
                                                                    ; Edited 13-Jan-88 19:23 by bvm
   (PROG (STREAM INFOBLK EOL)
         (AND OLDSTREAM (RETURN OLDSTREAM))
```

;; From REOPENFILE. Core files can't go away over logout, so just return old stream

```
         (COND
            [(type? STREAM NAME)
             (COND
                [(NULL (fetch ACCESS of NAME))
```

;; A closed file to be re-opened by its stream

```
                 (SETQ INFOBLK (fetch INFOBLK of NAME))
                 [if (EQ ACCESS 'OUTPUT)
                     then                                           ; Open for OUTPUT implies no content, so smash any existing
                                                                    ; pages
                          (replace IOEOFFSET of INFOBLK with 0)
                          (replace IOEPAGE of INFOBLK with 0)
                          (replace IOFILEPAGES of INFOBLK with (LIST (create CORE.PAGEENTRY
                                                                             PAGENUMBER _ 0]
                 (SETQ STREAM (create CORESTREAM smashing NAME DEVICE _ FDEV INFOBLK _ INFOBLK FULLFILENAME _
                                                          (fetch IOFILEFULLNAME of INFOBLK)
                                                          EOFFSET _ (fetch IOEOFFSET of INFOBLK)
                                                          EPAGE _ (fetch IOEPAGE of INFOBLK)
                                                          EOLCONVENTION _ (fetch COREEOLC of INFOBLK)
                                                          CBUFMAXSIZE _ BYTESPERPAGE OTHERPROPS _
                                                          (fetch OTHERPROPS of NAME]
                (((\IOMODEP NAME ACCESS T)
```

;; hdj - need we ever worry about being passed an already-open stream?

```
                 (RETURN NAME))
                (T (\FILE.WONT.OPEN NAME]
           [(SETQ STREAM (\COREFILE.GETFILEHANDLE NAME RECOG FDEV ACCESS))
            (COND
               ((NEQ ACCESS 'INPUT)
                (\COREFILE.SETPARAMETERS STREAM PARAMETERS))
               ((SETQ EOL (ASSOC 'EOL PARAMETERS))                  ; Set EOL for the input stream, in contradiction of whatever the
                                                                    ; file might have said.
                (replace EOLCONVENTION of STREAM with (SELECTQ (CADR EOL)
                                                               ((CR NIL)
                                                                    ; default
                                                                CR.EOLC)
```

```
                                                                (LF LF.EOLC)
                                                                (CRLF CRLF.EOLC)
                                                                (\ILLEGAL.ARG EOL]
            (T ;; Head for not-found error in \OPENFILE
               (RETURN NIL)))
          (\CORE.SETACCESSTIME STREAM ACCESS)
          (RETURN STREAM])
```

(\COREFILE.SETPARAMETERS

```
[LAMBDA (STREAM PARAMETERS)                                     ; Edited 5-Nov-87 17:50 by sye
  (for PAIR in PARAMETERS bind (INFOBLK _ (fetch INFOBLK of STREAM))
                               (TYPEFLG _ NIL)
     do (SELECTQ (CAR (LISTP PAIR))
            (EOL [replace EOLCONVENTION of STREAM with (replace COREEOLC of INFOBLK with (SELECTQ (CADR PAIR)
                                                                                                  ((CR NIL)
                                                                ; default
                                                                                                       CR.EOLC)
                                                                                                  (LF LF.EOLC)
                                                                                                  (CRLF CRLF.EOLC)
                                                                                                  (\ILLEGAL.ARG PAIR])
            ((TYPE FILETYPE)
                 (SETQ TYPEFLG T)
                 (replace IOFIBType of INFOBLK with (CADR PAIR)))
            ((CREATIONDATE ICREATIONDATE)
                 [replace IOFIBCreationTime of INFOBLK with (OR [FIXP (COND
                                                                        ((EQ (CAR PAIR)
                                                                             'CREATIONDATE)
                                                                         (IDATE (CADR PAIR)))
                                                                        (T (CADR PAIR]
                                                                (\ILLEGAL.ARG (CADR PAIR])
            NIL)
     finally (OR (fetch IOFIBType of INFOBLK)
                 TYPEFLG
                 (replace IOFIBType of INFOBLK with DEFAULTFILETYPE])
```

(\CORE.PACKFILENAME

```
[LAMBDA (DEVICE)
  (DECLARE (USEDFREE ROOT EXT VERS))                            ; Edited 13-Jan-88 19:42 by bvm
  (LET ((FULLNAME (CONCAT '{ (fetch (FDEV DEVICENAME) of DEVICE)
                          '} ROOT '%. EXT '; VERS)))
       (if *UPPER-CASE-FILE-NAMES*
           then (MKATOM (U-CASE FULLNAME))
         else FULLNAME])
```

(\CORE.RELEASEPAGES

```
[LAMBDA (STREAM LP)                                             (* rmk%: "23-SEP-83 16:02")
                                                                ; Release all pages of the file beyond the last page
  (OR LP (SETQ LP (fetch EPAGE of STREAM)))
  (for P in (fetch FILEPAGES of STREAM) when (ILESSP LP (fetch PAGENUMBER of P))
     do (replace PAGEPOINTER of P with NIL])
```

(\CORE.SETFILEPTR

```
[LAMBDA (STREAM INDX)                                           (* bvm%: " 9-Jul-84 14:25")
  (\CORE.UPDATEOF STREAM)                                       ; Update the EOF in case we have writen thru it
  (PROG ((NEWPAGE (fetch (BYTEPTR PAGE) of INDX))
         (NEWOFF (fetch (BYTEPTR OFFSET) of INDX)))
        (UNINTERRUPTABLY
            (COND
               ([OR (NEQ NEWPAGE (fetch CPAGE of STREAM))
                    (AND (APPENDONLY STREAM)
                         (ILESSP NEWOFF (fetch COFFSET of STREAM]
                                                                ; Force page release if ptr is going off the beaten path
                (replace CBUFSIZE of STREAM with 0)
                (replace CBUFPTR of STREAM with NIL)
                (replace CPAGE of STREAM with NEWPAGE)))
            (replace COFFSET of STREAM with NEWOFF))])
```

(\CORE.UPDATEOF

```
[LAMBDA (STREAM)                                                (* bvm%: " 9-Jul-84 14:25")
```

;; The EOF needs updating if we have written past the EOF. We check CBUFPTR to detect phony file positions from SETFILEPTR and
;; TURNPAGE that were never actually written thru

```
  (COND
     ([AND (fetch CBUFPTR of STREAM)
           (PROGN ;; Determines if the current file ptr is BEYOND the end of file. Since page is loaded, we can test against the CBUFSIZE. As
                  ;; we are ignoring the equal case, we dont need the test for page numbers used by FASTEOF.
                  (IGREATERP (fetch COFFSET of STREAM)
                             (fetch CBUFSIZE of STREAM]
      (UNINTERRUPTABLY
          (PROG ((OFF (fetch COFFSET of STREAM)))
```

```
                (COND
                   ((IGEQ OFF BYTESPERPAGE)
                    (add (fetch CPAGE of STREAM)
                         (fetch (BYTEPTR PAGE) of OFF))
                    (replace COFFSET of STREAM with (SETQ OFF (fetch (BYTEPTR OFFSET) of OFF)))
                    (replace CBUFPTR of STREAM with NIL)))
                (replace EPAGE of STREAM with (fetch CPAGE of STREAM))
                (replace EOFFSET of STREAM with OFF)
                (replace CBUFSIZE of STREAM with OFF)))])
```

(\CORE.BACKFILEPTR

```
(\CORE.BACKFILEPTR
  [LAMBDA (STREAM)                                          ; Edited  5-Nov-87 16:58 by sye
                                                            ; also see similar function \DRIBBACKFILEPTR
    [COND
       ((APPENDONLY STREAM)
        (LISPERROR "ILLEGAL ARG" (fetch (STREAM FULLNAME) of STREAM]
                                                            ; Checks done separately so we dont take an error with
                                                            ; interrupts off
    (COND
       ((NOT (AND (EQ (fetch COFFSET of STREAM)
                      0)
                  (EQ (fetch CPAGE of STREAM)
                      0)))
        (\CORE.UPDATEOF STREAM)
        (UNINTERRUPTABLY
            [replace COFFSET of STREAM with (COND
                                              ((EQ (fetch COFFSET of STREAM)
                                                   0)
                                               (replace CBUFSIZE of STREAM with 0)
                                               (replace CBUFPTR of STREAM with NIL)
                                               (add (fetch CPAGE of STREAM)
                                                    -1)
                                               (SUB1 BYTESPERPAGE))
                                              (T (SUB1 (fetch COFFSET of STREAM]
            [replace (STREAM CHARPOSITION) of STREAM with (IMAX 0 (SUB1 (fetch (STREAM CHARPOSITION) of STREAM])])
```

(\CORE.SETEOFPTR

```
(\CORE.SETEOFPTR
  [LAMBDA (STREAM NBYTES)                                   (* bvm%: "13-Feb-85 23:26")
    (\CORE.UPDATEOF STREAM)
    (PROG [(NEWBYTES (IDIFFERENCE NBYTES (\GETEOFPTR STREAM]
          (RETURN (COND
                     ((EQ NEWBYTES 0)                       ; Nothing to do
                      T)
                     ((OVERWRITEABLE STREAM)
                      (UNINTERRUPTABLY
                          [PROG ((NEWEP (fetch (BYTEPTR PAGE) of NBYTES))
                                 (NEWEO (fetch (BYTEPTR OFFSET) of NBYTES)))
                                (replace EPAGE of STREAM with NEWEP)
                                (replace EOFFSET of STREAM with NEWEO)
                                (replace CBUFSIZE of STREAM with (COND
                                                                   ((EQ NEWEP (fetch CPAGE of STREAM))
                                                                    NEWEO)
                                                                   (T (replace CBUFPTR of STREAM with NIL)
                                                                   ; Unmap noncurrent page
                                                                      0)))
                                (COND
                                   ((ILESSP NEWBYTES 0)         ; File is shorter
                                    (\ZEROBYTES (\CORE.FINDPAGE STREAM NEWEP)
                                           NEWEO
                                           (SUB1 BYTESPERPAGE)) ; Zero out the trailing fragment of the last page
                                    (\CORE.RELEASEPAGES STREAM NEWEP])
                      T])
```

(\CORE.SETACCESSTIME

```
(\CORE.SETACCESSTIME
  [LAMBDA (STREAM ACCESS)                                   (* rmk%: "23-SEP-83 14:38")
    ;; Set the 'last read' and/or 'last written' times for a core file according to access.
    (PROG ((DT (IDATE)))
          (SELECTQ ACCESS
               (INPUT (replace ReadTime of STREAM with DT))
               (BOTH (replace ReadTime of STREAM with DT)
                     (replace WriteTime of STREAM with DT))
               ((OUTPUT APPEND)
                    (replace WriteTime of STREAM with DT))
               (SHOULDNT)))
    STREAM])
```

(\CORE.SETFILEINFO

```
(\CORE.SETFILEINFO
  [LAMBDA (STREAM ATTRIBUTE VALUE DEV)
    ;; Edited  3-Jan-2022 20:00 by rmk: fixed bug--coercing CREATIONDATE twice   ; Edited  3-Jan-2022 19:59 by rmk
    ;; Edited 22-Nov-2021 09:25 by rmk:
                                                            (* bvm%: "15-Jan-85 17:40")
    (LET ((INFOBLOCK (\CORE.GETINFOBLOCK STREAM 'OLD DEV)))
```

```
        (AND INFOBLOCK (SELECTQ ATTRIBUTE
                            ((TYPE FILETYPE)
                                 (replace IOFIBType of INFOBLOCK with VALUE))
                            (EOL (replace COREEOLC of INFOBLOCK with (SELECTQ VALUE
                                                                             (CR CR.EOLC)
                                                                             (LF LF.EOLC)
                                                                             (CRLF CRLF.EOLC)
                                                                             (LISPERROR "ILLEGAL ARG" VALUE))))
                            (CREATIONDATE (replace IOFIBCreationTime of INFOBLOCK with (OR (IDATE VALUE)
                                                                                           (\ILLEGAL.ARG VALUE))))
                            (READDATE (replace IOFIBReadTime of INFOBLOCK with (OR (IDATE VALUE)
                                                                                   (\ILLEGAL.ARG VALUE))))
                            (WRITEDATE (replace IOFIBWriteTime of INFOBLOCK with (OR (IDATE VALUE)
                                                                                     (\ILLEGAL.ARG VALUE))))
                            (ICREATIONDATE
                                 (replace IOFIBCreationTime of INFOBLOCK with VALUE))
                            (IREADDATE (replace IOFIBReadTime of INFOBLOCK with VALUE))
                            (IWRITEDATE (replace IOFIBWriteTime of INFOBLOCK with VALUE))
                           NIL])
```

(\CORE.GETNEXTBUFFER

```
[LAMBDA (STREAM WHATFOR NOERRORFLG)                          ; Edited 17-Sep-90 13:22 by jds
```

;; Advances STREAM to a new page. Leaves the current page pointer NIL as the new page may never be written, so must update eof. Returns T
;; on success; any other return is a value to use by \BIN

```
(PROG ((CPAGE# (fetch CPAGE of STREAM))
       (COFF (fetch COFFSET of STREAM))
       EPAGE# COREBUF)
      [COND
         ((NOT (OPENED STREAM))
          (LISPERROR "FILE NOT OPEN" (fetch (STREAM FULLNAME) of STREAM]
      (if (AND (ILESSP COFF (SELECTQ WHATFOR
                                (READ (fetch CBUFSIZE of STREAM))
                               BYTESPERPAGE))
               (fetch CBUFPTR of STREAM))
          then                                               ; all OK, why were we called?
               (SHOULDNT)
               (RETURN T))
```

;; Buffer exhausted or empty, prepare new one

```
      (UNINTERRUPTABLY                                       ; Clean up current page
          (replace CBUFSIZE of STREAM with 0)
          (replace CBUFPTR of STREAM with NIL)
          (if (EQ COFF BYTESPERPAGE)
              then                                           ; Change to be first byte of next page instead of beyond last byte
                                                             ; of previous page
                   (replace COFFSET of STREAM with (SETQ COFF 0))
                   (replace CPAGE of STREAM with (add CPAGE# 1))))
      [COND
         ([AND (IGEQ CPAGE# (SETQ EPAGE# (fetch EPAGE of STREAM)))
               (OR (NEQ CPAGE# EPAGE#)
                   (IGEQ COFF (fetch EOFFSET of STREAM]       ; Current file pointer is at or past end of file
          (SELECTQ WHATFOR
               (READ (RETURN (AND (NULL NOERRORFLG)
                                  (\EOF.ACTION STREAM))))
               (WRITE (UNINTERRUPTABLY
                          (replace EPAGE of STREAM with (SETQ EPAGE# CPAGE#))
                          (replace EOFFSET of STREAM with COFF)))
               (\ILLEGAL.ARG WHATFOR]
```

;; Now fill the buffer -- map in current page

```
      (SETQ COREBUF (\CORE.FINDPAGE STREAM CPAGE#))          ; This is interruptable
      (UNINTERRUPTABLY                                       ; But these two fields must be set uninterruptably for benefit of
                                                             ; ucode
          (replace CBUFSIZE of STREAM with (COND
                                               ((ILESSP CPAGE# EPAGE#)
                                                                ; Full page
                                                BYTESPERPAGE)
                                               ((EQ EPAGE# CPAGE#)
                                                                ; Last page
                                                (fetch EOFFSET of STREAM))
                                               (T               ; Beyond EOF so no data
                                                  0)))
          (replace CBUFPTR of STREAM with COREBUF))
      (COND
         (\INTERRUPTABLE (BLOCK)))                           ; Let someone else run. Useful for those long writings of scratch
                                                             ; files.
      (RETURN T])
```

(\CORE.UNPACKFILENAME

```
[LAMBDA (NAME)                                               ; Edited 10-Jan-2022 22:42 by rmk
```

;; rmk; Convert / in ROOT to < or >   ; Edited 10-Jan-2022 21:14 by rmk

;; Edited 3-Nov-87 12:12 by bvm:

;; Breaks up a file name atom into its fields which it sets freely in its caller

```
    (DECLARE (USEDFREE ROOT EXT VERS))
    (PROG ((START (OR (AND (EQ (NTHCHAR NAME 1)
                               '{)
                           (STRPOS '} NAME NIL NIL NIL T))
                      1))
           (END (ADD1 (NCHARS NAME)))
           DOT SEMI)
          (SETQ DOT (STRPOS "." NAME START))
          (SETQ SEMI (OR (STRPOS ";" NAME DOT)
                         END))
          (COND
             ((NULL DOT)
              (SETQ DOT SEMI)))
          (SETQ ROOT (OR (SUBSTRING NAME START (SUB1 DOT))
                         ""))
          (CL:WHEN (STRPOS "/" ROOT)
              ;; If ROOT has slashes, convert to < ..> ..>
              (SETQ ROOT (DSUBST (CHARCODE >)
                                 (CHARCODE /)
                                 (CHCON ROOT)))
              (CL:WHEN (EQ (CAR ROOT)
                           (CHARCODE >))
                  (RPLACA ROOT (CHARCODE <)))
              (SETQ ROOT (CONCATCODES ROOT)))
          (SETQ EXT (COND
                       ((< DOT (- SEMI 1))
                        (SUBSTRING NAME (ADD1 DOT)
                                   (SUB1 SEMI)))
                       (T                                          ; null extension.
                          "")))
          (SETQ VERS (AND (< SEMI (- END 1))
                          (OR (FIXP (SUBATOM NAME (ADD1 SEMI)))
                              (CL:ERROR 'XCL:INVALID-PATHNAME :PATHNAME NAME])

)

(DEFINEQ

(COREDEVICE
  [LAMBDA (NAME NODIRFLG)                                          (* rmk%: " 1-NOV-83 18:34")
    (\DEFINEDEVICE NAME (\CREATECOREDEVICE NAME NODIRFLG])

(\CREATECOREDEVICE
  [LAMBDA (NAME NODIRFLG)                                          ; Edited 14-Feb-99 13:57 by rmk:
                                                                   ; Edited 19-Feb-93 15:57 by jds
    ;; DIRECTORYNAMEP has to be fixed up.  HOSTNAMEP is OK, cause each different host is defined by its own name.  Creates a NODIRCORE
    ;; device if NODIRFLG
    (create FDEV
            FDBINABLE _ T
            FDBOUTABLE _ T
            FDEXTENDABLE _ T
            DEVICENAME _ NAME
            RESETABLE _ T
            RANDOMACCESSP _ T
            PAGEMAPPED _ NIL
            NODIRECTORIES _ T
            BUFFERED _ T
            CLOSEFILE _ (FUNCTION \CORE.CLOSEFILE)
            DELETEFILE _ (COND
                            (NODIRFLG (FUNCTION NILL))
                            (T (FUNCTION \CORE.DELETEFILE)))
            GETFILEINFO _ (FUNCTION \CORE.GETFILEINFO)
            OPENFILE _ (COND
                          (NODIRFLG (FUNCTION \NODIRCORE.OPENFILE))
                          (T (FUNCTION \CORE.OPENFILE)))
            READPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
            SETFILEINFO _ (FUNCTION \CORE.SETFILEINFO)
            TRUNCATEFILE _ (FUNCTION \CORE.RELEASEPAGES)
            WRITEPAGES _ (FUNCTION \ILLEGAL.DEVICEOP)
            GETFILENAME _ (COND
                             (NODIRFLG (FUNCTION NILL))
                             (T (FUNCTION \CORE.GETFILENAME)))
            REOPENFILE _ (COND
                            [NODIRFLG (FUNCTION (LAMBDA (NAME ACCESS RECOG PARAMETERS FDEV OLDSTREAM)
                                                  OLDSTREAM]
                            (T (FUNCTION \CORE.OPENFILE)))
            GENERATEFILES _ (COND
                               (NODIRFLG (FUNCTION \NULLFILEGENERATOR))
                               (T (FUNCTION \CORE.GENERATEFILES)))
            EVENTFN _ (FUNCTION NILL)
            DEVICEINFO _ (AND (NOT NODIRFLG)
                              (LIST 'CoreFiles))
            DIRECTORYNAMEP _ (COND
```

```
                              (NODIRFLG (FUNCTION NILL))
                              (T                                    (* #.(SEDIT::MAKE-BROKEN-ATOM "WAS:") FUNCTION
                                                                    TRUE)
                                 (FUNCTION \CORE.DIRECTORYNAMEP)))
          HOSTNAMEP _ (FUNCTION NILL)
          READP _ (FUNCTION \GENERIC.READP)
          BIN _ (FUNCTION \BUFFERED.BIN)
          BOUT _ (FUNCTION \BUFFERED.BOUT)
          PEEKBIN _ (FUNCTION \BUFFERED.PEEKBIN)
          BACKFILEPTR _ (FUNCTION \CORE.BACKFILEPTR)
          SETFILEPTR _ (FUNCTION \CORE.SETFILEPTR)
          GETFILEPTR _ (FUNCTION \PAGEDGETFILEPTR)
          GETEOFPTR _ (FUNCTION \PAGEDGETEOFPTR)
          SETEOFPTR _ (FUNCTION \CORE.SETEOFPTR)
          EOFP _ (FUNCTION \PAGEDEOFP)
          BLOCKIN _ (FUNCTION \BUFFERED.BINS)
          BLOCKOUT _ (FUNCTION \BUFFERED.BOUTS)
          FORCEOUTPUT _ (FUNCTION NILL)
          GETNEXTBUFFER _ (FUNCTION \CORE.GETNEXTBUFFER)
          OPENP _ (FUNCTION \GENERIC.OPENP)
          REGISTERFILE _ (COND
                             (NODIRFLG (FUNCTION NILL))
                             (T (FUNCTION \ADD-OPEN-STREAM)))
          UNREGISTERFILE _ (COND
                             (NODIRFLG (FUNCTION NILL))
                             (T (FUNCTION \GENERIC-UNREGISTER-STREAM])

)

(DEFINEQ
```

(**NODIRCOREFDEV**

```
  [LAMBDA (NAME READPFN)                                            (* rmk%: " 1-NOV-83 18:33")
```

;; Creates a core device with no directory structure--files can't be found from names, only by saving a pointer to the stream. This is used for ;; linebuffers and perhaps other internal printing. The essential property is that the stream gets collected when it is no longer referenced.

```
    (PROG ((FDEV (\CREATECOREDEVICE NAME T)))
          (AND READPFN (replace READP of FDEV with READPFN))
          (\DEFINEDEVICE NAME FDEV)
          (RETURN FDEV])
```

(**NODIRCORE.OPENFILE**

```
  [LAMBDA (NAME ACCESS RECOG PARAMETERS FDEV)                       ; Edited  4-Jun-2022 16:27 by rmk
                                                                    (* lmm "24-May-85 11:59")
                                                                    ; Open function for NODIRCORE
    (COND
       [(type? STREAM NAME)
        (COND
           ((fetch ACCESS of NAME)
            (OR (\IOMODEP NAME ACCESS T)
                (\FILE.WONT.OPEN NAME)))
           (T (PROG ((INFOBLK (fetch INFOBLK of NAME)))
```

;; We'll return the stream that was given us, but we make sure that all its fields are back to their initial settings

```
                 (create CORESTREAM smashing NAME DEVICE _ FDEV INFOBLK _ INFOBLK FULLFILENAME _
                                             (fetch IOFILEFULLNAME of INFOBLK)
                                             EOFFSET _ (fetch IOEOFFSET of INFOBLK)
                                             EPAGE _ (fetch IOEPAGE of INFOBLK)
                                             EOLCONVENTION _ (fetch COREEOLC of INFOBLK)
                                             CBUFMAXSIZE _ BYTESPERPAGE OTHERPROPS _ (fetch OTHERPROPS
                                                                                        of NAME]
       (T (SELECTQ RECOG
                ((NEW OLD/NEW)
                     (SETQ NAME (create CORESTREAM
                                        DEVICE _ FDEV
                                        INFOBLK _ (create COREFILEINFOBLK)
                                        CBUFMAXSIZE _ BYTESPERPAGE)))
                (\FILE.WONT.OPEN NAME))
          (\COREFILE.SETPARAMETERS NAME PARAMETERS)))
    (\CORE.SETACCESSTIME NAME ACCESS)
    NAME])

)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(RECORD CORE.PAGEENTRY (PAGENUMBER . PAGEPOINTER))

(DATATYPE COREFILEINFOBLK ((IOFIBCreationTime FIXP)
                           (IOFIBReadTime FIXP)
                           (IOFIBWriteTime FIXP)
                           (IOFIBType POINTER)
                           (IOFILEPAGES POINTER)
                           (IOFILEFULLNAME POINTER)
```

```
                            (IOEPAGE WORD)
                            (IOEOFFSET WORD)
                            (COREEOLC BITS 2)
                            (IOFIBFileType WORD))
       IOFIBCreationTime _ (IDATE)
       IOFILEPAGES _ (LIST (create CORE.PAGEENTRY
                                   PAGENUMBER _ 0))
       COREEOLC _ CR.EOLC)

[RECORD CORESTREAM STREAM (SUBRECORD STREAM)
       (ACCESSFNS CORESTREAM ((INFOBLK (fetch F1 of DATUM)
                                       (replace F1 of DATUM with NEWVALUE))
                              (COREPAGECACHE (fetch F10 of DATUM)
                                       (replace F10 of DATUM with NEWVALUE))
                              (BEINGPRINTED (fetch IOBEINGPRINTED of (fetch INFOBLK of DATUM))
                                       (replace IOBEINGPRINTED of (fetch INFOBLK of DATUM) with NEWVALUE))
                              (FILEPAGES (fetch IOFILEPAGES of (fetch INFOBLK of DATUM))
                                       (replace IOFILEPAGES of (fetch INFOBLK of DATUM) with NEWVALUE))
                              (CreationTime (fetch IOFIBCreationTime of (fetch INFOBLK of DATUM))
                                       (replace IOFIBCreationTime of (fetch INFOBLK of DATUM) with NEWVALUE))
                              (ReadTime (fetch IOFIBReadTime of (fetch INFOBLK of DATUM))
                                       (replace IOFIBReadTime of (fetch INFOBLK of DATUM) with NEWVALUE))
                              (WriteTime (fetch IOFIBWriteTime of (fetch INFOBLK of DATUM))
                                       (replace IOFIBWriteTime of (fetch INFOBLK of DATUM) with NEWVALUE]

[ACCESSFNS COREDEVICE ((COREDIRECTORY (FETCH DEVICEINFO OF DATUM)
                              (REPLACE DEVICEINFO OF DATUM WITH NEWVALUE]

(RECORD COREGENFILESTATE (COREFILELST))
)

(/DECLAREDATATYPE 'COREFILEINFOBLK '(FIXP FIXP FIXP POINTER POINTER POINTER WORD WORD (BITS 2)
                                         WORD)
       ;; ---field descriptor list elided by lister---
       '16)
)

(/DECLAREDATATYPE 'COREFILEINFOBLK '(FIXP FIXP FIXP POINTER POINTER POINTER WORD WORD (BITS 2)
                                         WORD)
       ;; ---field descriptor list elided by lister---
       '16)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(COREDEVICE 'NODIRCORE T)

(COREDEVICE 'CORE)

(COREDEVICE 'SCRATCH T)
)

(DECLARE%: DOEVAL@LOAD DONTCOPY

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(LOCALVARS . T)
)
)

(PUTPROPS COREIO COPYRIGHT ("Venue & Xerox Corporation" 1981 1982 1983 1984 1985 1986 1987 1988 1990 1993 1999
                                 2018))
```

FUNCTION INDEX

RECORD INDEX