

File created: 23-Jan-2022 12:32:16 {DSK}<Users>kaplan>Local>medley3.5>my-medley>sources>CMLFILESYS.;
4

changes to: (FUNCTIONS CL:DIRECTORY)
previous date: 22-Jan-2022 09:26:49 {DSK}<Users>kaplan>Local>medley3.5>my-medley>sources>CMLFILESYS.;3
Read Table: XCL
Package: INTERLISP
Format: XCCS

; Copyright (c) 1986-1988, 1990 by Venue & Xerox Corporation.

```
(RPAQQ CMLFILESYSOMS ((FUNCTIONS CL:DIRECTORY CL:FILE-AUTHOR CL:FILE-LENGTH CL:FILE-POSITION  
CL:USER-HOMEDIR-PATHNAME CL:FILE-WRITE-DATE)  
(FUNCTIONS CL:PROBE-FILE CL:RENAME-FILE CL:DELETE-FILE)  
(PROP FILETYPE CMLFILESYS)))
```

```
(CL:DEFUN CL:DIRECTORY (PATHNAME &KEY CL::DEFAULTTEXT CL::DEFAULTVERS)  
; Edited 23-Jan-2022 12:32 by rmk  
; Edited 22-Jan-2022 09:26 by rmk  
(LET (GENERATOR FILE)  
(DECLARE (CL:SPECIAL GENERATOR))  
(RESETLST  
(CL:WHEN (EQL \\MACHINETYPE \\MAIKO)  
(RETSAVE NIL '(AND RESETSTATE (\\UFS.ABORT.CL-DIRECTORY))))  
(CL:SETQ GENERATOR (\\GENERATEFILES (DIRECTORY.FILL.PATTERN (CL:NAMESTRING PATHNAME)  
CL::DEFAULTTEXT CL::DEFAULTVERS)  
NIL  
'(SORT RESETLST)))  
(|while| (SETQ FILE (\\GENERATENEXTFILE GENERATOR)) |collect| (PATHNAME FILE))))))
```

```
(CL:DEFUN CL:FILE-AUTHOR (CL::FILE)
```

;;; Returns author of file as string, or NIL if it cannot be determined. FILE is a filename or stream.

```
(LET ((CL::AUTHOR (GETFILEINFO CL::FILE 'AUTHOR)))  
(CL:IF CL::AUTHOR  
(COERCE CL::AUTHOR 'CL:SIMPLE-STRING)  
NIL)))
```

```
(CL:DEFUN CL:FILE-LENGTH (FILE-STREAM)  
(|if| (AND (STREAMP FILE-STREAM)  
(OPENP FILE-STREAM))  
|then| (GETEOFPTR FILE-STREAM)))
```

```
(CL:DEFUN CL:FILE-POSITION (CL::FILE-STREAM &OPTIONAL (CL:POSITION NIL CL::POSITIONP))  
(CL:UNLESS (STREAMP CL::FILE-STREAM)  
(\\ILLEGAL.ARG CL::FILE-STREAM))  
(CL:IF CL::POSITIONP  
(CL:IF (RANDACCESSP CL::FILE-STREAM)  
(PROGN (SETFILEPTR CL::FILE-STREAM (CASE CL:POSITION  
(:START 0)  
(:END (GETEOFPTR CL::FILE-STREAM))  
(T CL:POSITION)))  
T)  
NIL)  
(GETFILEPTR CL::FILE-STREAM)))
```

```
(CL:DEFUN CL:USER-HOMEDIR-PATHNAME (&OPTIONAL HOST)  
(DECLARE (GLOBALVARS LOGINHOST/DIR *DEFAULT-PATHNAME-DEFAULTS*))  
(CL:IF (MACHINETYPE 'MAIKO)  
(CL:IF (AND HOST (CL:STRING-NOT-EQUAL (STRING HOST)  
(UNIX-GETPARM "HOSTNAME")))  
NIL  
(CL:MAKE-PATHNAME :HOST :DSK :DIRECTORY (UNPACKFILENAME.STRING (UNIX-GETENV "HOME")  
'DIRECTORY  
'RETURN)))  
(PATHNAME (OR LOGINHOST/DIR *DEFAULT-PATHNAME-DEFAULTS*))))
```

```
(CL:DEFUN CL:FILE-WRITE-DATE (FILE)
```

;; Return file's creation date, or NIL if it doesn't exist.

;; N.B. date is returned in Common Lisp Universal Time, not Interlisp-D internal time

```
(LET ((TN (CL:PROBE-FILE FILE)))  
(CL:WHEN TN  
(%CONVERT-INTERNAL-TIME-TO-CLUT (GETFILEINFO TN 'ICREATIONDATE))))))
```

(CL:DEFUN **CL:PROBE-FILE** (FILE)

::: Return a pathname which is the truename of the file if it exists, NIL otherwise. Returns NIL for non-file args.

```

(IF (STREAMP FILE)
  THEN (IF (OPENP FILE)
    THEN (PATHNAME (FETCH (STREAM FULLNAME) OF FILE))
    ELSE (LET ((NAMESTRING-IF-EXISTS (INFILEP (FETCH (STREAM FULLNAME) OF FILE))))
      (AND NAMESTRING-IF-EXISTS (PATHNAME NAMESTRING-IF-EXISTS))))
  ELSE (LET ((INFILEP (\\GETFILENAME FILE 'OLD)))
    (IF INFILEP
      THEN (PATHNAME INFILEP)
      ELSE NIL))))

```

(CL:DEFUN **CL:RENAME-FILE** (FILE NEW-NAME)

::: Give FILE the new name NEW-NAME. If FILE is an open stream, error. Otherwise, do the rename. If successful, return three values: the new name, truename of original file, truename of new file.

```

(LET ((OLD-PATHNAME (PATHNAME FILE))
      (CL::NEW-FULLNAME))
  (IF (STREAMP FILE)
    THEN (IF (OPENP FILE)
      THEN (CL:ERROR "Renaming open streams is not supported: ~S" FILE)
      ELSE (SETQ CL::NEW-FULLNAME (RENAMEFILE (SETQ FILE (FETCH (STREAM FULLNAME) OF FILE))
        NEW-NAME)))
    ELSE (SETQ CL::NEW-FULLNAME (RENAMEFILE FILE NEW-NAME)))
  (IF CL::NEW-FULLNAME
    THEN (CL:VALUES (CL:MERGE-PATHNAMES NEW-NAME FILE)
      OLD-PATHNAME
      (PATHNAME CL::NEW-FULLNAME))
    ELSE (CL:ERROR "Rename failed"))))

```

(CL:DEFUN **CL:DELETE-FILE** (FILE)

(* * "Delete the specified file.")

```

(LET ((TN (CL:PROBE-FILE FILE)))
  (CL:WHEN (STREAMP FILE)
    (CL:CLOSE FILE :ABORT T))
  (CL:IF TN
    (LET ((NS (INTERLISP-NAMESTRING TN)))
      (CL:UNLESS (DELFILE NS)
        (CL:ERROR "Could not delete the file ~S" FILE)))
    (CL:UNLESS (STREAMP FILE)
      (CL:ERROR "File to be deleted does not exist: ~S" FILE)))
  T)

```

(PUTPROPS **CMLFILESYS FILETYPE** CL:COMPILE-FILE)

(PUTPROPS **CMLFILESYS COPYRIGHT** ("Venue & Xerox Corporation" 1986 1987 1988 1990))

FUNCTION INDEX

CL:DELETE-FILE	2	CL:FILE-LENGTH	1	CL:PROBE-FILE	2
CL:DIRECTORY	1	CL:FILE-POSITION	1	CL:RENAME-FILE	2
CL:FILE-AUTHOR	1	CL:FILE-WRITE-DATE	1	CL:USER-HOMEDIR-PATHNAME	1

PROPERTY INDEX

CMLFILESYS	2
------------------	---
