

File created: 27-Jul-90 07:53:18 {DSK}<usr>local>lde>loops>src>SYSTEM>LOOPS-INSTALL.;2

changes to: (VARS LOOPS-INSTALLCOMS)

previous date: 1-Dec-88 12:14:30 {DSK}<usr>local>lde>loops>src>SYSTEM>LOOPS-INSTALL.;1

Read Table: XCL

Package: INTERLISP

Format: XCCS

; Copyright (c) 1988, 1990 by Venue & Xerox Corporation. All rights reserved.

```
(RPAQQ LOOPS-INSTALLCOMS ((FNS LOAD-SYSTEM LOAD-SYSTEM1 LS-FIX-DIRS LS-FLOPPY-PROMPT LS-GRAPHIC LS-INSTALL
LS-LOAD LS-LOAD? LS-MAKE-FLOPPIES LS-PROMPT LS-PROMPT-DONE LS-SET-DIRS)))
```

```
(DEFINEQ
```

```
(LOAD-SYSTEM
```

```
(LAMBDA (DESC)
```

```
; Edited 6-Jul-88 15:22 by raf
```

```
;;; Just calls LOAD-SYSTEM1 in a separate process to avoid possibly screwing ourselves...
```

```
(OR DESC (SETQ DESC LS-CURRENT-SYSTEM))
(ADD.PROCESS `(PROGN (CL:FORMAT T "Current system is ~A~%" ',(CL:FIRST DESC))
(GIVE.TTY.PROCESS (\\INSUREWINDOW (TTYDISPLAYSTREAM)))
(CL:UNWIND-PROTECT
  (LOAD-SYSTEM1 ',DESC)
  (CLOSEW (TTYDISPLAYSTREAM))))
'NAME
(CONCAT "Installing " (CL:FIRST LS-CURRENT-SYSTEM))
'BEFOREEXIT
'DON\('T))
```

```
(LOAD-SYSTEM1
```

```
(LAMBDA (DESC)
```

```
; Edited 13-Jul-88 16:39 by raf
```

```
;;; DESC should be a list of four elements:
```

```
;; NAME - a string that names the system
;; DIRECTORIES - a list of atoms and default values which will be set to the directories/subdirectories where the system is going to live, like this:
;; ((LOOPSDIRECTORY {DSK}<LISPFILES>LOOPS>) (LOOPSLIBRARY {DSK}<LISPFILES>LOOPS>LIBRARY>))
;; INSTALL-LIST - a list of lists of descriptions of floppies, each of which looks like this:
;; ("Source Floppy 5" (LOOPSDIRECTORY foo bar baz)(LOOPSLIBRARY mumble frotz))
;; There is one list per (MACHINETYPE).
;; LOAD-LIST - a list of things that can be loaded once stuff is installed, along with code to load them: ((LOOPS (FILESLOAD FROM
;; LOOPSDIRECTORY foo bar baz))(MUMBLE (FILESLOAD FROM LOOPSLIBRARY mumble))
;; Spelling correction is OFF here; DWIM once bit me by turning a COPYFILES into a COPYFILE in an expression that was trying to load
;; COPYFILES...
```

```
(LET* ((LS-CURRENT-SYSTEM (OR DESC LS-CURRENT-SYSTEM))
      (LS-MENU (|create| MENU
                TITLE _ (CL:FORMAT NIL "~A System" (CL:FIRST LS-CURRENT-SYSTEM))
                ITEMS _ `((, (|if| (EQ (MACHINETYPE)
                                      'MAIKO)
                                |then| "Install from distribution"
                                |else| "Install from floppies")
                          'INSTALL
                          , (|if| (EQ (MACHINETYPE)
                                      'MAIKO)
                                |then| "Gets local directories and builds LOOPSSITE file"
                                |else| "Gets local directories, copies from floppies and builds LOOPSSITE
                                      file"))
                          ("Load into sysout" 'LOAD "Loads installed system into sysout")))
                MENUPOSITION _ (CONS (IQUOTIENT SCREENHEIGHT 2)
                                     (IQUOTIENT SCREENWIDTH 2))))
      (NOSPELLFLG T))
(DECLARE (SPECVARS LS-CURRENT-SYSTEM NOSPELLFLG))
(|while| (SELECTQ (MENU LS-MENU)
                (INSTALL (LS-SET-DIRS (CL:FIRST LS-CURRENT-SYSTEM)
                                     (CL:SECOND LS-CURRENT-SYSTEM)
                                     'ALWAYS)
                          (LS-INSTALL (CL:THIRD LS-CURRENT-SYSTEM)
                                     T)
                          (LOAD (LS-SET-DIRS (CL:FIRST LS-CURRENT-SYSTEM)
                                             (CL:SECOND LS-CURRENT-SYSTEM))
                                (LS-LOAD (CL:FOURTH LS-CURRENT-SYSTEM)
                                         T)
                                NIL)
                          |do| NIL))))
```

```
(LS-FIX-DIRS
```

```
(LAMBDA (ITEM WINDOW BUTTONS) ; Edited 21-Mar-88 16:28 by jrb:
  (LET (|something-changed|)
    (|for| D |in| (WINDOWPROP WINDOW 'DIRLIST) |bind| FMI FMVAL
      |do| (SETQ FMI (FM.GETITEM (CAR D)
        'DIRNAMES WINDOW))
        (SETQ FMVAL (FM.ITEMPROP FMI 'LABEL))
        (|if| (OR (NOT (BOUNDP (CAR D)))
          (NOT (EQUAL FMVAL (FM.ITEMPROP FMI 'INITSTATE))))
          |then| (SET (CAR D)
            (MKATOM FMVAL))
            (SETQ |something-changed| T)))
        (|if| |something-changed|
          |then| (EVAL (WINDOWPROP WINDOW 'DIRFORMS)))
        (NOTIFY.EVENT (WINDOWPROP WINDOW 'DONEVENT))
        (CLOSEW WINDOW))))
```

(LS-FLOPPY-PROMPT

```
(LAMBDA (NAME DONT-CHECK) ; Edited 6-Apr-88 19:41 by BANE
  (|while| T |bind| BOGUSNAME |do| (LS-PROMPT (CONCAT "Please insert floppy " NAME))
    ;; This is necessary to reliably detect a floppy swap on an 1186...GRRRR!
    (\\FLOPPY.CLOSE)
    (|if| (OR DONT-CHECK (EQUAL (SETQ BOGUSNAME (FLOPPY.NAME))
      NAME))
      |then| (RETURN)
      |else| (CL:FORMAT T "~%That was ~A!~%" BOGUSNAME)
        (RINGBELLS))))
```

(LS-GRAPHIC

```
(LAMBDA (ITEM WINDOW CHAR) ; Edited 1-Dec-88 12:10 by bane
  ;; The first AND clause catches the digits 0-9 which FreeMenu passes in as SMALLPs.
  (OR (AND (SMALLP CHAR)
    (IGREATERP CHAR -1)
    (ILESSP CHAR 10))
    (LET ((CHAROBJ (COERCE CHAR 'CL:CHARACTER)))
      (AND (CL:GRAPHIC-CHAR-P CHAROBJ)
        (NOT (EQ CHAROBJ #\Space))))))
```

(LS-INSTALL

```
(LAMBDA (INSDISC) ; Edited 6-Jul-88 16:09 by raf
  (LET ((INSLIST (CAR INSDISC))
    FLOPPYRECS)
    (SELECTQ (MACHINETYPE)
      ((DANDELION DOVE)
        (SETQ FLOPPYRECS (ASSOC (MACHINETYPE)
          INSLIST))
        (|if| FLOPPYRECS
          |then| (|for| FLOPPYREC |in| (CDR FLOPPYRECS)
            |do| (|if| (EQ (CAR FLOPPYREC)
              'EVAL)
              |then| (EVAL (CADR FLOPPYREC))
              |else| (LS-FLOPPY-PROMPT (CAR FLOPPYREC))
                (|for| DIRREC |in| (CDR FLOPPYREC)
                  |do| (CL:FORMAT T "~&Connecting to ~A" (EVAL (CAR DIRREC)))
                    (CNDIR (EVAL (CAR DIRREC)))
                    (|for| FNAME |in| (CDR DIRREC)
                      |do| (CL:FORMAT T "~&Copying ~A" FNAME)
                        (COPYFILE (CL:FORMAT NIL "{FLOPPY}~A" FNAME)
                          FNAME)))
                    (CL:FORMAT T "~&Done with ~A" (CAR FLOPPYREC))))
                |else| (CL:BREAK "Can't find distribution description for ~A" (MACHINETYPE))))
          (DORADO (CL:WARN "DORADOes do not have floppy drives."))
          (MAIKO (CL:WARN "You should already have used TAR to unload the distribution tape."))
          (CL:BREAK "Unrecognized machine type ~s" (MACHINETYPE)))
      (EVAL (CADR INSDISC))))
```

(LS-LOAD

```
(LAMBDA (LOADLIST) ; Edited 4-Apr-88 20:31 by jrb:
  ;; LOADLIST can be:
  ;; A list: it is treated as a list of load command lists and a menu lets the user select which one to load
  ;; NIL: the load command list from LS-CURRENT-SYSTEM is used as above
  ;; Otherwise: it is treated as the option to select from the load command list from LS-CURRENT-SYSTEM
  (OR LOADLIST (SETQ LOADLIST (CL:FOURTH LS-CURRENT-SYSTEM)))
  (LET ((WHICHLL (|for| LLE |in| LOADLIST
    |bind| (SELECTED _ (MENU (|create| MENU
      TITLE _ "Load Which?"
      ITEMS _ (|for| I |in| LOADLIST
        |collect| (|if| (CL:SYMBOLP (CAR I))
          |then| (CAR I)
          |else| (LIST (CAAR I)
            (CL:FORMAT T "~A" (CAR I))))))))
```

```

(MENUPOSITION _ (CONS (IQUOTIENT SCREENHEIGHT 2)
                      (IQUOTIENT SCREENWIDTH 2))))
(CAAR I)
(CADAR I)))
|when| (OR (AND (CL:SYMBOLP (CAR LLE))
              (EQUAL SELECTED (CAR LLE)))
          (AND (CL:CONSP (CAR LLE))
              (EQUAL SELECTED (CAAR LLE))))
|do| (RETURN LLE)))
(|for| FORM |in| (CDR WHICHLL) |do| (EVAL FORM))))

```

(LS-LOAD?)

; Edited 30-Mar-88 14:40 by jrb:

```

(LAMBDA (FILENAME WHICH-FLOPPY)
  (|if| (NOT (GETPROP FILENAME 'FILEDATES))
    |then| (|if| (NOT (APPLY* 'FILESLOAD ' (NOERROR)
                          FILENAME))
      |then| (|while| (NOT (GETPROP FILENAME 'FILEDATES))
        |do| (CL:FORMAT T "You need to load file ~A and I can't find it~%" FILENAME)
              (LS-FLOPPY-PROMPT WHICH-FLOPPY)
              (|if| (NOT (APPLY* 'FILESLOAD ' (FROM {FLOPPY})
                                FILENAME))
                |then| (CL:FORMAT T "~A wasn't on that floppy; check it and try again~%"))))
        (CL:FORMAT T "Loaded file ~A~%" FILENAME))))

```

(LS-MAKE-FLOPPIES)

; Edited 6-Apr-88 17:42 by BANE

```

(LAMBDA NIL
  (LET ((INSTALL-LIST (CAR (CL:THIRD LS-CURRENT-SYSTEM)))
        (LS-SET-DIRS (CL:FIRST LS-CURRENT-SYSTEM)
                     (CL:SECOND LS-CURRENT-SYSTEM)
                     'ALWAYS)
        (SELECTQ (MACHINETYPE)
                  ((DANDELION DOVE)
                   (|for| FLOPPYREC |in| (CDR (ASSOC (MACHINETYPE)
                                                       INSTALL-LIST))
                     |unless| (EQ (CAR FLOPPYREC)
                                  'EVAL)
                     |do| (LS-FLOPPY-PROMPT (CAR FLOPPYREC)
                                             T)
                           (|for| DIRREC |in| (CDR FLOPPYREC) |do| (CL:FORMAT T "~&Connecting to ~A"
                                                                              (EVAL (CAR DIRREC)))
                                                                    (CNDIR (EVAL (CAR DIRREC)))
                                                                    (|for| FNAME |in| (CDR DIRREC)
                                                                      |do| (CL:FORMAT T "~&Copying ~A" FNAME)
                                                                              (COPYFILE FNAME (CL:FORMAT NIL "{FLOPPY}~A"
                                                                              FNAME))))))
                   (CL:FORMAT T "~&Done with ~A" (CAR FLOPPYREC))))
        (CL:BREAK "This machine has no floppies I know about; you're on your own..."))))

```

(LS-PROMPT)

; Edited 1-Apr-88 20:44 by jrb:

```

(LAMBDA (TITLE-STRING)
  ;; A persistent prompt that won't go away
  (LET ((FM (FREEMENU `((LABEL ,TITLE-STRING HJUSTIFY CENTER)
                       (PROPS ID DONEBUTTON)
                       (LABEL "Click here when done" HJUSTIFY CENTER SELECTEDFN LS-PROMPT-DONE))))))
    (WINDOWPROP FM 'DONEBUTTON (CREATE.EVENT))
    (MOVEW FM (IQUOTIENT SCREENHEIGHT 2)
             (IQUOTIENT SCREENWIDTH 2))
    (OPENW FM)
    (AWAIT.EVENT (WINDOWPROP FM 'DONEBUTTON))))

```

(LS-PROMPT-DONE)

; Edited 1-Apr-88 20:30 by jrb:

```

(LAMBDA (ITEM WINDOW BUTTONS)
  (NOTIFY.EVENT (WINDOWPROP WINDOW 'DONEBUTTON))
  (CLOSEW WINDOW))

```

(LS-SET-DIRS)

; Edited 4-Apr-88 20:08 by jrb:

```

(LAMBDA (NAME DIRDESC ALWAYS?)
  ;; Don't make the window if all the directory variables are already bound - Do do it if ALWAYS? is true
  ;; Structure of DIRDESC is:
  ;; (((DIRATOM1 "{directory}<path>name1>" ...)
  ;; (forms to run after getting DIRATOMs set))
  (LET ((DIRLIST (CAR DIRDESC))
        (OR
         (AND (NOT ALWAYS?)
              (|for| D |in| DIRLIST |always| (BOUNDP (CAR D))))))
        (LET ((DIRMENU
              (FREEMENU `((PROPS FORMAT COLUMN)
                          (PROPS ID DIRATOMS)

```

```
(LABEL ,(CL:FORMAT NIL "~A directories" NAME)
      TYPE DISPLAY HJUSTIFY LEFT)
,@(|for| D |in| DIRLIST |collect| `(LABEL ,(CAR D)
                                         TYPE DISPLAY)))
((PROPS ID DIRNAMES)
 (LABEL "Click here when done" HJUSTIFY RIGHT SELECTEDFN LS-FIX-DIRS)
 ,@(|for| D |in| DIRLIST
   |collect| `(LABEL ,(OR (AND (BOUNDP (CAR D))
                              (EVAL (CAR D)))
                          (CADR D))
             ID
             ,(CAR D)
             TYPE EDIT LIMITCHARS LS-GRAPHIC))))))
(WINDOWPROP DIRMENU 'DIRLIST DIRLIST)
(WINDOWPROP DIRMENU 'DIRFORMS (CADR DIRDESC))
(WINDOWPROP DIRMENU 'DONEVENT (CREATE.EVENT))
(GETMOUSESTATE)
(MOVEW DIRMENU (IQUOTIENT SCREENHEIGHT 2)
          (IQUOTIENT SCREENWIDTH 2))
(OPENW DIRMENU)
(AWAIT.EVENT (WINDOWPROP DIRMENU 'DONEVENT))))))
```

```
)
(PUTPROPS LOOPS-INSTALL COPYRIGHT ("Venue & Xerox Corporation" 1988 1990))
```

FUNCTION INDEX

LOAD-SYSTEM	1	LS-FLOPPY-PROMPT	2	LS-LOAD	2	LS-PROMPT	3
LOAD-SYSTEM1	1	LS-GRAPHIC	2	LS-LOAD?	3	LS-PROMPT-DONE	3
LS-FIX-DIRS	1	LS-INSTALL	2	LS-MAKE-FLOPPIES	3	LS-SET-DIRS	3
