```
;;
;; Copyright (c) 1986, 1987, 1988 by Xerox Corporation.  All rights reserved.
```

```
(RPAQQ LOOPS-ABCCOMS
        ((COMS
```

;;; The Loops tester

```
        ;; This stuff is bogus, and should be removed
        [INITVARS (LOOPSTESTDIR "{POGO:}<LOOPS>TESTER>")
                (CoreTesterFiles '(LTBASIC LTSUB LTKER LTFILES LTCASES))
                (TESTFILES '(LTBASIC LTSUB LTKER LTFILES LTCASES LTCORETESTS LTLISPFUNTESTS LTMETHODTESTS
                                LTPUTMETHODTEST]
        (FNS TESTLOOPS))
```

;;; Loops debugging stuff

```
        (INITVARS (\DebugLoopsOldDir NIL))
        (FNS AnalyzeLoops DebugLoops LOOPSDIR i/d PPI)
        (ADDVARS (*GIVE-AND-TAKE-DIRECTORIES* "{POGO:}<LOOPSCORE>SYSTEM>"))
        (FILES LOOPS-EXPORTS.ALL)
        (DECLARE%: DONTCOPY (PROP MAKEFILE-ENVIRONMENT LOOPS-ABC))))
```

;;; The Loops tester
;; This stuff is bogus, and should be removed

```
(RPAQ? LOOPSTESTDIR "{POGO:}<LOOPS>TESTER>")
```

```
(RPAQ? CoreTesterFiles '(LTBASIC LTSUB LTKER LTFILES LTCASES))
```

```
(RPAQ? TESTFILES '(LTBASIC LTSUB LTKER LTFILES LTCASES LTCORETESTS LTLISPFUNTESTS LTMETHODTESTS LTPUTMETHODTEST)
)
```

```
(DEFINEQ
```

```
(TESTLOOPS
  [LAMBDA (options breakFlg)                                    (* smL "22-Apr-86 19:19")

          (* * loads and runs tests as determined by the options)

          (* * options -
          T%: Load LTFILES and stop. NIL -
          load LTFILES and then load StandardTestFiles and run them. varName -
          load LTFILES and then load files in varName and run them)

    (SETQ DIRECTORIES (UNION (LIST LOOPSTESTDIR LOOPSDIRECTORY)
                            DIRECTORIES))
    (DOFILESLOAD CoreTesterFiles)
    (COND
       ((EQ options T))
       (T (TestFromFiles (COND
                            ((NULL options)
                             AllTestFiles)
                            ((LITATOM options)
                             (EVAL options))
                            (T options))
                breakFlg])
)
```

;;; Loops debugging stuff

```
(RPAQ? \DebugLoopsOldDir NIL)
```

```
(DEFINEQ
```

```
(AnalyzeLoops
  [LAMBDA (updateDB?)                                          ; Edited 22-Jun-87 09:39 by smL
```

;;; Analyze all the Loops files with Masterscope.

```
        (DebugLoops)
        (if updateDB?
            then [for file in LOOPSFILES do (for fn in (FILECOMSLST file)
                                                  do (MASTERSCOPE '(ANALYZE ',fn]
          elseif (AND (MOUSECONFIRM "Delete current MasterScope database and analyze Loops from scratch?" NIL
                            \TopLevelTtyWindow T)
                      (MOUSECONFIRM "Are you sure?" NIL \TopLevelTtyWindow T))
            then (MASTERSCOPE '(ERASE))
                 (for file in LOOPSFILES do (printout T "Loading fns from " file "...")
                                            (LOADFNS (for fn in (FILECOMSLST file)
                                                         when [NOT (OR (EXPRP fn)
                                                                       (EXPRP (GETPROP fn 'EXPR]
                                                         collect fn)
                                                     file
                                                     'PROP)
                                            (printout T "Analyzing")
                                            [for fn in (FILECOMSLST file)
                                                do (MASTERSCOPE '(ANALYZE ',fn]
                                            (printout T T])
```

(**DebugLoops**
```
  [LAMBDA (turnOffFlg)                                                           ; Edited 25-Jun-87 12:51 by smL

    ;; Add LOOPSFILES to FILELST and LOOPSDIRECTORY to directories, or reverse if turnOffFlg =T.  Used to put Loops into a mode for debugging
    ;; Loops system code.

    (DECLARE (GLOBALVARS \DebugLoopsOldDir))
    (if turnOffFlg
        then (SETQ FILELST (for x in FILELST when [AND (NOT (FMEMB x LOOPSFILES))
                                                       (NOT (MEMB x '(LOOPS-EXPORTS.ALL LOOPS-EXPORTS LOOPS-ABC]
                                  collect x))
             (SETQ DIRECTORIES (for x in DIRECTORIES when (NEQ x LOOPSDIRECTORY) collect x))
             (COND
                (\DebugLoopsOldDir (CNDIR \DebugLoopsOldDir)
                        (SETQ \DebugLoopsOldDir NIL)))
             "Return to normal"
      else (SETQ DIRECTORIES (UNION (LIST LOOPSDIRECTORY)
                                    DIRECTORIES))
           (if (NULL \DebugLoopsOldDir)
               then (SETQ \DebugLoopsOldDir (DIRECTORYNAME T T)))
           (LOOPSDIR)
           (FILESLOAD LOOPS-EXPORTS.ALL)
           (for f in LOOPSFILES unless (BOUNDP (FILECOMS f)) do (LOADFROM f NIL 'ALLPROP))
           (SETQ FILELST (UNION (for file in LOOPSFILES collect file when (GETPROP file 'FILE))
                                FILELST))
           "Debugging Loops"])
```

(**LOOPSDIR**
```
  [LAMBDA (SUBDIR)                                                            (* smL "27-Feb-85 13:24")

           (* * Connects to the directory for saving LOOPS sources.)

    (/CNDIR (PACK* LOOPSDIRECTORY (OR SUBDIR ""]
```

(**i/d**
```
  [LAMBDA (item)                                                             (* dgb%: " 3-DEC-82 01:46")
                                                                             (* short form of call)
    (INSPECT/DATATYPE item])
```

(**PPI**
```
  [LAMBDA (INSTANCE RECORDNAME FILE)                                         (* smL "17-Jan-86 16:32")
                                                                             (* Pretty-prints an instance of a record.)
    (PROG [(POS (ADD1 (POSITION FILE)))
           (DEC (RECLOOK (OR RECORDNAME (COND
                                          ((LISTP INSTANCE)
                                           (CAR INSTANCE))
                                          (T (TYPENAME INSTANCE]
          (COND
             (DEC (printout FILE "[" %# (for FIELD in (RECORDFIELDNAMES DEC) unless (EQ FIELD 'PERFORMOPS)
                                            do (printout NIL .TAB0 POS .P2 FIELD " = " .PPV (RECORDACCESS FIELD
                                                                                               INSTANCE DEC))
                                 )
                       "]" T))
             (T (printout FILE .PPV INSTANCE T])
)

(ADDTOVAR *GIVE-AND-TAKE-DIRECTORIES* "{POGO:}<LOOPSCORE>SYSTEM>")

(FILESLOAD LOOPS-EXPORTS.ALL)

(DECLARE%: DONTCOPY
```

(PUTPROPS **LOOPS-ABC MAKEFILE-ENVIRONMENT** (:PACKAGE "IL" :READTABLE "INTERLISP" :BASE 10))
)

(PUTPROPS **LOOPS-ABC COPYRIGHT** ("Xerox Corporation" 1986 1987 1988))

FUNCTION INDEX

VARIABLE INDEX

PROPERTY INDEX