

File created: 25-Oct-91 16:28:50 {DSK}<usr>local>lde>lispcore>sources>CONDITION-HIERARCHY.;3

changes to: (IL:VARS IL:CONDITION-HIERARCHYCOMS)

previous date: 30-Jul-91 15:14:18 {DSK}<usr>local>lde>lispcore>sources>CONDITION-HIERARCHY.;2

Read Table: XCL

Package: XEROX-COMMON-LISP

Format: XCCS

; Copyright (c) 1986, 1987, 1988, 1990, 1991 by Venue & Xerox Corporation. All rights reserved.

```
(IL:RPAQQ IL:CONDITION-HIERARCHYCOMS
  ((IL:STRUCTURES CONDITION SIMPLE-CONDITION WARNING SIMPLE-WARNING SERIOUS-CONDITION ERROR SIMPLE-ERROR
    ASSERTION-FAILED HASH-TABLE-FULL)
  (IL:STRUCTURES CELL-ERROR UNBOUND-VARIABLE UNDEFINED-FUNCTION ATTEMPT-TO-CHANGE-CONSTANT
    ATTEMPT-TO-RPLAC-NIL)
  (IL:FILES IL:CONDITION-HIERARCHY-SI IL:CONDITION-HIERARCHY-POST-SI)
  (IL:COMS (IL:FUNCTIONS IL:PRETTY-TYPE-NAME)
    (IL:STRUCTURES TYPE-ERROR SIMPLE-TYPE-ERROR TYPE-MISMATCH SYMBOL-AS-PATHNAME))
  (IL:STRUCTURES CONTROL-ERROR PROGRAM-ERROR ILLEGAL-GO ILLEGAL-RETURN ILLEGAL-THROW BAD-PROCEED-CASE)
  (IL:STRUCTURES STREAM-ERROR STREAM-NOT-OPEN READ-ERROR SYMBOL-NAME-TOO-LONG END-OF-FILE)
  (IL:STRUCTURES STORAGE-CONDITION STACK-OVERFLOW CRITICAL-STORAGE-CONDITION STORAGE-EXHAUSTED
    SYMBOL-HT-FULL ARRAY-SPACE-FULL DATA-TYPES-EXHAUSTED)
  (IL:STRUCTURES DEVICE-ERROR SIMPLE-DEVICE-ERROR)
  (IL:STRUCTURES FILE-ERROR FILE-WONT-OPEN FS-RESOURCES-EXCEEDED FS-PROTECTION-VIOLATION
    FS-RENAMEFILE-SOURCE-COULDNT-DELETE)
  (IL:STRUCTURES ARITHMETIC-ERROR DIVISION-BY-ZERO FLOATING-POINT-OVERFLOW FLOATING-POINT-UNDERFLOW)
  (IL:STRUCTURES PATHNAME-ERROR FILE-NOT-FOUND INVALID-PATHNAME)
  (IL:STRUCTURES CL::PRINT-NOT-READABLE)
  (IL:FUNCTIONS SIMPLE-CONDITION-FORMAT-ARGUMENTS SIMPLE-CONDITION-FORMAT-STRING)
  (IL:FILES IL:CONDITION-HIERARCHY-IL)
  (IL:PROP (IL:FILETYPE IL:MAKEFILE-ENVIRONMENT)
    IL:CONDITION-HIERARCHY)))

(DEFINE-CONDITION CONDITION NIL
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Condition ~S occurred." CONDITION))))

(DEFINE-CONDITION SIMPLE-CONDITION (CONDITION)
  (FORMAT-STRING FORMAT-ARGUMENTS)
  (:CONC-NAME "%SIMPLE-CONDITION-")
  (:REPORT (LAMBDA (CONDITION STREAM)
    (APPLY 'FORMAT STREAM (%SIMPLE-CONDITION-FORMAT-STRING CONDITION)
      (%SIMPLE-CONDITION-FORMAT-ARGUMENTS CONDITION)))))

(DEFINE-CONDITION WARNING (CONDITION)
  (CONDITION)
  (:REPORT (LAMBDA (C S)
    (FORMAT S "Warning: ~A" (WARNING-CONDITION S)))))

(DEFINE-CONDITION SIMPLE-WARNING (WARNING)
  (FORMAT-STRING FORMAT-ARGUMENTS)
  (:CONC-NAME "%SIMPLE-WARNING-")
  (:REPORT (LAMBDA (CONDITION STREAM)
    (APPLY 'FORMAT STREAM (%SIMPLE-WARNING-FORMAT-STRING CONDITION)
      (%SIMPLE-WARNING-FORMAT-ARGUMENTS CONDITION)))))

(DEFINE-CONDITION SERIOUS-CONDITION (CONDITION)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Serious condition ~S occurred." (TYPE-OF CONDITION)))))

(DEFINE-CONDITION ERROR (SERIOUS-CONDITION)
  NIL)

(DEFINE-CONDITION SIMPLE-ERROR (ERROR)
  (FORMAT-STRING FORMAT-ARGUMENTS)
  (:CONC-NAME "%SIMPLE-ERROR-")
  (:REPORT (LAMBDA (CONDITION STREAM)
    (APPLY 'FORMAT STREAM (%SIMPLE-ERROR-FORMAT-STRING CONDITION)
      (%SIMPLE-ERROR-FORMAT-ARGUMENTS CONDITION)))))

(DEFINE-CONDITION ASSERTION-FAILED (SIMPLE-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (APPLY 'FORMAT T (OR (ASSERTION-FAILED-FORMAT-STRING CONDITION)
```

"Assertion failed."
(ASSERTION-FAILED-FORMAT-ARGUMENTS CONDITION)))))

(DEFINE-CONDITION **HASH-TABLE-FULL** (ERROR)
(TABLE)
(:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
(FORMAT T "Hash table full: ~S" (HASH-TABLE-FULL-TABLE CONDITION)))))

(DEFINE-CONDITION **CELL-ERROR** (ERROR)
(NAME))

(DEFINE-CONDITION **UNBOUND-VARIABLE** (CELL-ERROR)
NIL
(:REPORT (LAMBDA (C S)
(FORMAT S "~S is an unbound variable." (UNBOUND-VARIABLE-NAME C)))))

(DEFINE-CONDITION **UNDEFINED-FUNCTION** (CELL-ERROR)
NIL
(:REPORT (LAMBDA (C S)
(FORMAT S "~S is an undefined function." (UNDEFINED-FUNCTION-NAME C)))))

(DEFINE-CONDITION **ATTEMPT-TO-CHANGE-CONSTANT** (CELL-ERROR)
NIL)

(DEFINE-CONDITION **ATTEMPT-TO-RPLAC-NIL** (ATTEMPT-TO-CHANGE-CONSTANT)
NIL
(:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
(FORMAT T "Attempt to rplac NIL with ~S" (ATTEMPT-TO-RPLAC-NIL-NAME CONDITION)))))

(IL:FILESLOAD IL:CONDITION-HIERARCHY-SI IL:CONDITION-HIERARCHY-POST-SI)

(DEFUN **IL:PRETTY-TYPE-NAME** (IL:TYPEPEC)
(IL:IF (EQ (CAR (IL:LISTP IL:TYPEPEC))
'OR)
IL:THEN (LET ((IL:TYPES (IL:SUBSET (CDR IL:TYPEPEC)
(IL:FUNCTION (IL:LAMBDA (IL:NAME)
(NOT (IL:SOME (CDR IL:TYPEPEC)
(IL:FUNCTION (IL:LAMBDA (IL:OTHER)
(AND (IL:NEQ IL:OTHER IL:NAME)
(SUBTYPEP IL:NAME IL:OTHER)
))))))))))
(IL:IF (IL:EQUAL (IL:SORT IL:TYPES)
'(COMPLEX FLOAT INTEGER RATIO))
IL:THEN "a number"
IL:ELSE (IL:CONCATLIST (CDR (IL:FOR IL:X IL:IN IL:TYPES IL:JOIN (LIST " or " (IL:PRETTY-TYPE-NAME
IL:X))))))
IL:ELSE (LET (IL:DOC)
(IF (AND (SYMBOLP IL:TYPEPEC)
(IL:SETQ IL:DOC (DOCUMENTATION IL:TYPEPEC 'TYPE)))
IL:DOC
(IL:CONCAT "a " IL:TYPEPEC))))))

(DEFINE-CONDITION **TYPE-ERROR** (ERROR)
(EXPECTED-TYPE DATUM)
(:REPORT (LAMBDA (C S)
(FORMAT S "Arg not ~A~&~S" (IL:PRETTY-TYPE-NAME (TYPE-ERROR-EXPECTED-TYPE C)
(TYPE-ERROR-DATUM C)))))

(DEFINE-CONDITION **SIMPLE-TYPE-ERROR**

;;; This is a pretty worthless type to have around.

(TYPE-ERROR)
(FORMAT-STRING FORMAT-ARGUMENTS)
(:CONC-NAME "%SIMPLE-TYPE-ERROR-")
(:REPORT (LAMBDA (C S)
(APPLY 'FORMAT S (%SIMPLE-TYPE-ERROR-FORMAT-STRING C)
(%SIMPLE-TYPE-ERROR-FORMAT-ARGUMENTS C)))))

(DEFINE-CONDITION **TYPE-MISMATCH** (TYPE-ERROR)
(NAME VALUE MESSAGE)
(:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
(IF (EQL (TYPE-MISMATCH-NAME CONDITION)
(TYPE-MISMATCH-VALUE CONDITION))
(FORMAT T "~S is not ~A." (TYPE-MISMATCH-VALUE CONDITION)
(OR (TYPE-MISMATCH-MESSAGE CONDITION)

```

      (IL:PRETTY-TYPE-NAME (TYPE-MISMATCH-EXPECTED-TYPE CONDITION)))
    (FORMAT T "The value of ~S, ~S, is not ~A." (TYPE-MISMATCH-NAME CONDITION)
      (TYPE-MISMATCH-VALUE CONDITION)
      (OR (TYPE-MISMATCH-MESSAGE CONDITION)
        (IL:PRETTY-TYPE-NAME (TYPE-MISMATCH-EXPECTED-TYPE CONDITION))))))

```

```

(DEFINE-CONDITION SYMBOL-AS-PATHNAME (TYPE-ERROR)
  (SYMBOL WHERE MESSAGE)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (IF (SYMBOL-AS-PATHNAME-MESSAGE CONDITION)
      (FORMAT T (SYMBOL-AS-PATHNAME-MESSAGE CONDITION)
        (SYMBOL-AS-PATHNAME-SYMBOL CONDITION))
      (FORMAT T "~a: symbol ~s used as pathname" (SYMBOL-AS-PATHNAME-WHERE CONDITION)
        (SYMBOL-AS-PATHNAME-SYMBOL CONDITION))))))

```

```

(DEFINE-CONDITION CONTROL-ERROR (ERROR)
  NIL)

```

```

(DEFINE-CONDITION PROGRAM-ERROR (ERROR)
  NIL)

```

```

(DEFINE-CONDITION ILLEGAL-GO (PROGRAM-ERROR)
  (TAG)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "GO to a nonexistent tag: ~S." (ILLEGAL-GO-TAG CONDITION))))))

```

```

(DEFINE-CONDITION ILLEGAL-RETURN (PROGRAM-ERROR)
  (TAG)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "RETURN to nonexistent block: ~S." (ILLEGAL-RETURN-TAG CONDITION))))))

```

```

(DEFINE-CONDITION ILLEGAL-THROW (CONTROL-ERROR)
  (TAG)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Tag for THROW not found: ~S." (ILLEGAL-THROW-TAG CONDITION))))))

```

```

(DEFINE-CONDITION BAD-PROCEED-CASE (CONTROL-ERROR)
  (NAME)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Proceed case ~S is not currently active." (BAD-PROCEED-CASE-NAME CONDITION))))))

```

```

(DEFINE-CONDITION STREAM-ERROR (ERROR)
  (STREAM)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Stream error on ~S." (STREAM-ERROR-STREAM CONDITION))))))

```

```

(DEFINE-CONDITION STREAM-NOT-OPEN (STREAM-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Stream not open: ~S" (STREAM-NOT-OPEN-STREAM CONDITION))))))

```

```

(DEFINE-CONDITION READ-ERROR (ERROR)
  NIL)

```

```

(DEFINE-CONDITION SYMBOL-NAME-TOO-LONG (READ-ERROR)
  NIL
  (:REPORT "Symbol name too long"))

```

```

(DEFINE-CONDITION END-OF-FILE (STREAM-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION STREAM)
    (FORMAT STREAM "End of file ~S" (END-OF-FILE-STREAM CONDITION))))))

```

```

(DEFINE-CONDITION STORAGE-CONDITION (SERIOUS-CONDITION)
  NIL)

```

```

(DEFINE-CONDITION STACK-OVERFLOW (STORAGE-CONDITION)
  NIL
  (:REPORT "Stack overflow"))

```

```

(DEFINE-CONDITION CRITICAL-STORAGE-CONDITION (STORAGE-CONDITION)
  NIL)

```

```
(DEFINE-CONDITION STORAGE-EXHAUSTED (CRITICAL-STORAGE-CONDITION)
  NIL)

(DEFINE-CONDITION SYMBOL-HT-FULL (CRITICAL-STORAGE-CONDITION)
  NIL
  (:REPORT "Symbol hash table full"))

(DEFINE-CONDITION ARRAY-SPACE-FULL (CRITICAL-STORAGE-CONDITION)
  NIL
  (:REPORT "Array space full"))

(DEFINE-CONDITION DATA-TYPES-EXHAUSTED (CRITICAL-STORAGE-CONDITION)
  NIL
  (:REPORT "No more data types available"))

(DEFINE-CONDITION DEVICE-ERROR (SERIOUS-CONDITION)
  (DEVICE))

(DEFINE-CONDITION SIMPLE-DEVICE-ERROR (DEVICE-ERROR)
  (MESSAGE)
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Device error: ~A" (SIMPLE-DEVICE-ERROR-MESSAGE CONDITION))))))

(DEFINE-CONDITION FILE-ERROR (ERROR)
  (PATHNAME))

(DEFINE-CONDITION FILE-WONT-OPEN (FILE-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "File won't open: ~A" (FILE-WONT-OPEN-PATHNAME CONDITION))))))

(DEFINE-CONDITION FS-RESOURCES-EXCEEDED (FILE-ERROR)
  NIL
  (:REPORT (LAMBDA (C S)
    (FORMAT S "File system resources exceeded: ~A" (FS-RESOURCES-EXCEEDED-PATHNAME C))))))

(DEFINE-CONDITION FS-PROTECTION-VIOLATION (FILE-ERROR)
  NIL
  (:REPORT (LAMBDA (C S)
    (FORMAT S "Protection violation: ~A" (FILE-ERROR-PATHNAME C))))))

(DEFINE-CONDITION FS-RENAMEFILE-SOURCE-COULDNT-DELETE (FILE-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Couldn't delete the source file: ~A" (FS-RENAMEFILE-SOURCE-COULDNT-DELETE-PATHNAME
      CONDITION))))))

(DEFINE-CONDITION ARITHMETIC-ERROR (ERROR)
  (OPERATION OPERANDS)
  (:REPORT (LAMBDA (C S)
    (FORMAT S "Arithmetic error during (~S~{ ~S~})" (ARITHMETIC-ERROR-OPERATION C)
      (ARITHMETIC-ERROR-OPERANDS C))))))

(DEFINE-CONDITION DIVISION-BY-ZERO (ARITHMETIC-ERROR)
  NIL
  (:REPORT "Attempt to divide by zero.))

(DEFINE-CONDITION FLOATING-POINT-OVERFLOW (ARITHMETIC-ERROR)
  NIL
  (:REPORT "Floating point overflow.))

(DEFINE-CONDITION FLOATING-POINT-UNDERFLOW (ARITHMETIC-ERROR)
  NIL
  (:REPORT "Floating point underflow.))

(DEFINE-CONDITION PATHNAME-ERROR (ERROR)
  (PATHNAME))

(DEFINE-CONDITION FILE-NOT-FOUND (FILE-ERROR)
```

```
NIL
(:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
  (FORMAT T "File not found: ~A" (FILE-NOT-FOUND-PATHNAME CONDITION))))
(:HANDLE (LAMBDA (CONDITION)
  (COND
    ((BOUNDP 'IL:ERRORPOS)
     (LET ((NEWNAME (IL:SPELLFILE (IL:ROOTFILENAME (FILE-NOT-FOUND-PATHNAME CONDITION))
                                   NIL IL:NOFILESPELLFLG)))
       (COND
         (NEWNAME (IL:ENVAPPLY (IL:STKNAME IL:ERRORPOS)
                               (IL:SUBST NEWNAME (FILE-NOT-FOUND-PATHNAME CONDITION)
                                             (MAPCAR #'(LAMBDA (X)
                                                       (IF (PATHNAMEP X)
                                                           (NAMESTRING X)
                                                           X)))
                               (IL:STKARGS IL:ERRORPOS)))
          (IL:STKNTH -1 IL:ERRORPOS IL:ERRORPOS)
          IL:ERRORPOS T T))))))))))
```

```
(DEFINE-CONDITION INVALID-PATHNAME (PATHNAME-ERROR)
  NIL
  (:REPORT (LAMBDA (CONDITION *STANDARD-OUTPUT*)
    (FORMAT T "Invalid pathname: ~A" (INVALID-PATHNAME-PATHNAME CONDITION))))))
```