

File created: 14-Jun-90 21:03:42 {DSK}<usr>local>lde>lispcore>internal>library>CONDITIONGRAPH.;2

changes to: (VARS CONDITIONGRAPHCOMS)

previous date: 9-Dec-87 16:48:03 {DSK}<usr>local>lde>lispcore>internal>library>CONDITIONGRAPH.;1

Read Table: XCL

Package: INTERLISP

Format: XCCS

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

```
(RPAQQ CONDITIONGRAPHCOMS ((DECLARE\ :DOEVAL@LOAD DOEVAL@COMPILE DONTCOPY (XCL:FILE-ENVIRONMENTS
:CONDITIONGRAPH))
(VARIABLES *CONDITION-GRAPH-WINDOW* *CONDITION-GRAPH-SEXPR*)
(FUNCTIONS EDIT-CONDITIONS GRAPH-CONDITIONS CONDITION-SUBGRAPH
CONDITION-SUBGRAPH-RECURSION RECOMPUTE-CONDITION-GRAPH-SEXPR
COUNT-CONDITION-TYPES COUNT-CONDITION-TYPES-RECURSION)
(PROP CONDITIONGRAPH)))
```

```
(DECLARE\ :DOEVAL@LOAD DOEVAL@COMPILE DONTCOPY
```

```
(XCL:DEFINE-FILE-ENVIRONMENT :CONDITIONGRAPH :READTABLE "XCL"
:PACKAGE "IL"
:COMPILER :COMPILE-FILE)
)
```

```
(CL:DEFVAR *CONDITION-GRAPH-WINDOW* NIL
"Window in which to display the condition hierarchy graph.")
```

```
(CL:DEFVAR *CONDITION-GRAPH-SEXPR* NIL
"Tree structure representing last calculated condition type graph.")
```

```
(CL:DEFUN EDIT-CONDITIONS (ROOT)
(CL:LABELS ((EDIT-CONDITIONS-RECURSION (GRAPH)
(CL:UNLESS (NULL GRAPH)
(ED (CL:FIRST GRAPH)
:STRUCTURES)
(CL:MAPC #'EDIT-CONDITIONS-RECURSION (CL:REST GRAPH))))))
(EDIT-CONDITIONS-RECURSION (CONDITION-SUBGRAPH ROOT NIL))))
```

```
(CL:DEFUN GRAPH-CONDITIONS (&OPTIONAL (ROOT 'CONDITION)
(RECOMPUTE (NULL *CONDITION-GRAPH-SEXPR*)
W)
(LET ((NEWW (SHOWGRAPH (LAYOUTSEXPR (CONDITION-SUBGRAPH ROOT RECOMPUTE)
' (HORIZONTAL))
(OR W *CONDITION-GRAPH-WINDOW* (CL:FORMAT NIL "Condition type graph from: ~S" ROOT))
NIL NIL T)))
(WINDOWPROP NEWW 'TITLE (CL:FORMAT NIL "Condition type graph from: ~S" ROOT))
(OR W *CONDITION-GRAPH-WINDOW* (CL:SETF *CONDITION-GRAPH-WINDOW* NEWW))))
```

```
(CL:DEFUN CONDITION-SUBGRAPH (ROOT RECOMPUTE &AUX (ONCE NIL)
RESULT)
(CL:UNLESS (CL:SUBTYPEP ROOT 'CONDITION)
(CL:ERROR "~S is not a condition type."))
(CL:LOOP (CL:WHEN RECOMPUTE (RECOMPUTE-CONDITION-GRAPH-SEXPR))
(CL:SETF RESULT (CONDITION-SUBGRAPH-RECURSION ROOT *CONDITION-GRAPH-SEXPR*))
(CL:WHEN (OR ONCE RESULT)
(CL:RETURN-FROM CONDITION-SUBGRAPH RESULT))
(CL:FORMAT *ERROR-OUTPUT* "Couldn't find ~S in current graph.")
(CL:SETQ ONCE T RECOMPUTE T)))
```

```
(CL:DEFUN CONDITION-SUBGRAPH-RECURSION (TARGET TREE)
(COND
(NULL TREE)
NIL)
(EQ TARGET (CL:FIRST TREE))
TREE)
(T (CL:DOLIST (SUBTREE (CL:REST TREE))
(LET ((FOUND? (CONDITION-SUBGRAPH-RECURSION TARGET SUBTREE)))
(CL:WHEN FOUND? (RETURN FOUND?))))))
```

```
(CL:DEFUN RECOMPUTE-CONDITION-GRAPH-SEXPR ()
(LET ((CGHASH (CL:MAKE-HASH-TABLE)))
(CL:FORMAT *ERROR-OUTPUT* " Computing condition hierarchy graph.")
(MAPCAR (DATATYPES)
#' (CL:LAMBDA (SYMBOL)
(BLOCK)
```

```

(CL:WHEN (AND (NOT (CL:GETHASH SYMBOL CGHASH))
              (CL:SUBTYPEP SYMBOL 'CONDITION))
  (CL:DO ((TYPE SYMBOL (CONDITION-PARENT TYPE))
          (CHAIN NIL))
    ((COND
      ((NULL TYPE)
       (CL:SETF *CONDITION-GRAPH-SEXPR* CHAIN))
      ((CL:GETHASH TYPE CGHASH)
       (NCONC (CL:GETHASH TYPE CGHASH)
              (LIST CHAIN)))
      (T NIL)))
    (CL:PRINC ".")
    (CL:SETF (CL:GETHASH TYPE CGHASH)
              (CL:SETF CHAIN (CL:IF (NULL CHAIN)
                                     (LIST TYPE)
                                     (LIST TYPE CHAIN))))))))))

```

```

(CL:DEFUN COUNT-CONDITION-TYPES ()
  (COUNT-CONDITION-TYPES-RECURSION (CONDITION-SUBGRAPH 'CONDITION NIL)))

```

```

(CL:DEFUN COUNT-CONDITION-TYPES-RECURSION (TREE)
  (COND
    ((NULL TREE)
     0)
    ((CL:SYMBOLP TREE)
     1)
    (T (FOR SUBTREE IN TREE SUM (COUNT-CONDITION-TYPES-RECURSION SUBTREE))))))

```

```

(PUTPROPS CONDITIONGRAPH COPYRIGHT ("Venue & Xerox Corporation" 1986 1987 1990))

```

**FUNCTION INDEX**

CONDITION-SUBGRAPH .....	1	COUNT-CONDITION-TYPES-RECURSION ..	2	RECOMPUTE-CONDITION-GRAPH-SEXPR ..	1
CONDITION-SUBGRAPH-RECURSION .....	1	EDIT-CONDITIONS .....	1		
COUNT-CONDITION-TYPES .....	2	GRAPH-CONDITIONS .....	1		

---

**VARIABLE INDEX**

*CONDITION-GRAPH-SEXPR* .....	1	*CONDITION-GRAPH-WINDOW* .....	1
-------------------------------	---	--------------------------------	---

---

**FILE-ENVIRONMENT INDEX**

:CONDITIONGRAPH .....	1
-----------------------	---

---