

File created: 24-Mar-92 14:06:32 {DSK}<usr>local>lde>lispcore>sources>SEdit-LIST-FORMATS.;2

changes to: (IL:SEdit-FORMATS LOCALLY)  
(IL:VARS IL:SEdit-LIST-FORMATSCOMS)

previous date: 17-May-90 11:08:07 {DSK}<usr>local>lde>lispcore>sources>SEdit-LIST-FORMATS.;1

Read Table: XCL

Package: SEDIT

Format: XCCS

; Copyright (c) 1987, 1988, 1990, 1992 by Venue & Xerox Corporation. All rights reserved.

```
(IL:RPAQQ IL:SEdit-LIST-FORMATSCOMS
  ((IL:PROP (IL:FILETYPE IL:MAKEFILE-ENVIRONMENT)
    IL:SEdit-LIST-FORMATS)
    (IL:DECLARE\ : IL:DONTCOPY IL:DOEVAL@COMPILE (IL:FILES IL:SEdit-INDENT))
    (IL:SEdit-FORMATS RETURN-FROM THROW BLOCK
      ;; some handy formats under names we don't expect to find as CAR of form
      :DEFAULT :DATA :BINDING :BINDING-LIST :LAMBDA-LIST :FBINDING :FBINDING-LIST :COND-CLAUSE
      :CASE-CLAUSE :HORIZONTAL :HORIZONTAL-NOBREAK :VERTICAL :VERTICAL-NOBREAK :KEYWORD-ARGS
      :KEYWORD-ARGS-NOBREAK :SQUASHED :DEDIT)
    (IL:SEdit-FORMATS
      ;; special forms & macros
      AND CASE CATCH CCASE CERROR COMPILER-LET COND CTYPECASE DECLARE DEFCommand DEFConstant
      DEFDefiner DEFun DEFINE-SPECIAL-FORM DEFInline DEFGlobalVar DEFGlobalParameter DEFList-Format
      DEFMacro DEFOptimizer DEFPackage DEFParameter DEFSetf DEFStruct DEFTYPE DEFVar
      DESTRUCTURING-BIND DO DO* DOLIST DOTIMES DO-ALL-SYMBOLS DO-EXTERNAL-SYMBOLS
      DO-INTERNAL-SYMBOLS DO-LOCAL-SYMBOLS DO-SYMBOLS ECASE ERROR ETYPECASE EVAL-WHEN FLET FORMAT IF
      IGNORE-ERRORS LABELS LAMBDA IL:LAMBDA LET LET* LOCALLY LOCALLY LIST LIST* MACROLET
      MULTIPLE-VALUE-BIND MULTIPLE-VALUE-SETQ IL:NLAMBDA IL:OPENLAMBDA OR IL:P PROCEED-CASE PROG
      PROG* PROGn PROGV RESTART-CASE IL:SELCHARQ IL:SELECTQ TAGBODY THE TYPECASE UNDOABLY UNLESS
      UNWIND-PROTECT WHEN WITH-INPUT-FROM-STRING WITH-OUTPUT-TO-STRING WITH-OPEN-FILE
      WITH-OPEN-STREAM IL:WITH.MONITOR IL:WITH.FAST.MONITOR IL:WITH.SPY)))

(IL:PUTPROPS IL:SEdit-LIST-FORMATS IL:FILETYPE :COMPILE-FILE)

(IL:PUTPROPS IL:SEdit-LIST-FORMATS IL:MAKEFILE-ENVIRONMENT NIL)

(IL:DECLARE\ : IL:DONTCOPY IL:DOEVAL@COMPILE

(IL:FILESLOAD IL:SEdit-INDENT)
)

(DEF-List-Format RETURN-FROM CATCH)

(DEF-List-Format THROW CATCH)

(DEF-List-Format BLOCK CATCH)

(DEF-List-Format :DEFAULT "Used for lists whose CAR has no specified format."
  :INDENT :HORIZONTAL-ATOM
  :INLINE T)

(DEF-List-Format :DATA "Used for quoted (not backquoted) lists."
  :INDENT :DATA
  :ARGS (:RECURSIVE)
  :INLINE T)

(DEF-List-Format :BINDING :INDENT :BINDING
  :INLINE T)

(DEF-List-Format :BINDING-LIST :INDENT :BINDING-LIST
  :ARGS (:BINDING)
  :INLINE NIL)

(DEF-List-Format :LAMBDA-LIST :INDENT :LAMBDA-LIST
  :ARGS (:BINDING)
  :INLINE T)

(DEF-List-Format :FBINDING :INDENT (:NOBREAK 1)
  :ARGS (NIL :LAMBDA-LIST NIL)
  :SUBLISTS (2))
```

```

{MEDLEY}<CLTL2>SEdit-LIST-FORMATS.;1

(DEF-LIST-FORMAT :FBINDING-LIST :INDENT :BINDING-LIST
  :ARGS (:FBINDING))

(DEF-LIST-FORMAT :COND-CLAUSE :INDENT :COND-CLAUSE
  :INLINE T)

(DEF-LIST-FORMAT :CASE-CLAUSE :INDENT :COND-CLAUSE
  :ARGS (:DATA NIL)
  :SUBLISTS (1)
  :INLINE T)

(DEF-LIST-FORMAT :HORIZONTAL :INDENT :HORIZONTAL
  :INLINE T)

(DEF-LIST-FORMAT :HORIZONTAL-NOBREAK :INDENT :HORIZONTAL
  :INLINE T
  :MISER :NEVER)

(DEF-LIST-FORMAT :VERTICAL :INDENT :VERTICAL
  :INLINE T)

(DEF-LIST-FORMAT :VERTICAL-NOBREAK :INDENT :VERTICAL
  :INLINE T
  :MISER :NEVER)

(DEF-LIST-FORMAT :KEYWORD-ARGS :INDENT :KEYWORD-ARG
  :INLINE T)

(DEF-LIST-FORMAT :KEYWORD-ARGS-NOBREAK :INDENT :KEYWORD-ARG
  :INLINE T
  :MISER :NEVER)

(DEF-LIST-FORMAT :SQUASHED :INDENT :SQUASH
  :INLINE T)

(DEF-LIST-FORMAT :DEDIT :SQUASHED)

(DEF-LIST-FORMAT AND :INDENT :VERTICAL
  :MISER :NEVER
  :INLINE T)

(DEF-LIST-FORMAT CASE :INDENT (1)
  :ARGS (NIL NIL :CASE-CLAUSE)
  :SUBLISTS T)

(DEF-LIST-FORMAT CATCH :INDENT (1)
  :ARGS (:KEYWORD NIL)
  :INLINE T)

(DEF-LIST-FORMAT CCASE CASE)

(DEF-LIST-FORMAT CERROR FORMAT)

(DEF-LIST-FORMAT COMPILER-LET LET)

(DEF-LIST-FORMAT COND :INDENT :VERTICAL
  :ARGS (NIL :COND-CLAUSE)
  :SUBLISTS T)

(DEF-LIST-FORMAT CTYPECASE TYPECASE)

(DEF-LIST-FORMAT DECLARE :INDENT :VERTICAL
  :ARGS (:KEYWORD :HORIZONTAL)
  :SUBLISTS T)

(DEF-LIST-FORMAT DEFCOMMAND DEFUN)

```

```
(DEF-LIST-FORMAT DEFCONSTANT DEFVAR)
```

```
(DEF-LIST-FORMAT DEFDEFINER :INDENT ((2)
                                         1)
  :ARGS (:KEYWORD :CASE-CLAUSE NIL :LAMBDA-LIST NIL))
```

```
(DEF-LIST-FORMAT DEFUN :INDENT ((2))
  :ARGS (:KEYWORD NIL :LAMBDA-LIST NIL)
  :SUBLISTS (3))
```

```
(DEF-LIST-FORMAT DEFINE-SPECIAL-FORM DEFUN)
```

```
(DEF-LIST-FORMAT DEFINLINE DEFUN)
```

```
(DEF-LIST-FORMAT DEFGLOBALVAR DEFVAR)
```

```
(DEF-LIST-FORMAT DEFGLOBALPARAMETER DEFVAR)
```

```
(DEF-LIST-FORMAT DEF-LIST-FORMAT :INDENT (1)
  :ARGS (:KEYWORD NIL)
  :INLINE T)
```

```
(DEF-LIST-FORMAT DEFMACRO DEFUN)
```

```
(DEF-LIST-FORMAT DEFOPTIMIZER
```

;;: note: this loses in the case where OPT-NAME is specified in DEFOPTIMIZER form because we can't handle optional arguments at this level. we  
 ;;: could go through the pain of writing a custom indent spec...

```
DEFUN)
```

```
(DEF-LIST-FORMAT DEFPACKAGE :INDENT (1)
  :ARGS (:KEYWORD NIL :CASE-CLAUSE)
  :INLINE NIL)
```

```
(DEF-LIST-FORMAT DEFPARAMETER DEFVAR)
```

```
(DEF-LIST-FORMAT DEFSETF :INDENT ((3))
  :ARGS (:KEYWORD NIL :LAMBDA-LIST :BINDING NIL)
  :SUBLISTS (3 4))
```

```
(DEF-LIST-FORMAT DEFSTRUCT :INDENT (1)
  :ARGS (:KEYWORD NIL))
```

```
(DEF-LIST-FORMAT DEFTYPE DEFMACRO)
```

```
(DEF-LIST-FORMAT DEFVAR :INDENT (1)
  :ARGS (:KEYWORD NIL)
  :INLINE T)
```

```
(DEF-LIST-FORMAT DESTRUCTURING-BIND :INDENT (1 1)
  :ARGS (:KEYWORD :LAMBDA-LIST NIL)
  :INLINE NIL)
```

```
(DEF-LIST-FORMAT DO :INDENT (:TAGBODY 2)
  :ARGS (:KEYWORD :BINDING-LIST :COND-CLAUSE NIL)
  :SUBLISTS (2 3))
```

```
(DEF-LIST-FORMAT DO* DO)
```

```
(DEF-LIST-FORMAT DOLIST :INDENT (:TAGBODY :STEP 1)
  :ARGS (:KEYWORD :BINDING NIL)
  :SUBLISTS (2))
```

```
(DEF-LIST-FORMAT DOTIMES DOLIST)
```

```
(DEF-LIST-FORMAT DO-ALL-SYMBOLS DOLIST)

(DEF-LIST-FORMAT DO-EXTERNAL-SYMBOLS DOLIST)

(DEF-LIST-FORMAT DO-INTERNAL-SYMBOLS DOLIST)

(DEF-LIST-FORMAT DO-LOCAL-SYMBOLS DOLIST)

(DEF-LIST-FORMAT DO-SYMBOLS DOLIST)

(DEF-LIST-FORMAT ECASE CASE)

(DEF-LIST-FORMAT ERROR FORMAT)

(DEF-LIST-FORMAT ETYPECASE TYPECASE)

(DEF-LIST-FORMAT EVAL-WHEN :INDENT (1)
  :SUBLISTS (2))

(DEF-LIST-FORMAT FLET :INDENT (1)
  :ARGS (:KEYWORD :FBINDING-LIST NIL)
  :SUBLISTS (2))

(DEF-LIST-FORMAT FORMAT :INDENT :HORIZONTAL
  :INLINE T)

(DEF-LIST-FORMAT IF :INDENT (2)
  :INLINE T)

(DEF-LIST-FORMAT IGNORE-ERRORS UNDOABLY)

(DEF-LIST-FORMAT LABELS FLET)

(DEF-LIST-FORMAT LAMBDA :INDENT (1)
  :ARGS (:KEYWORD :LAMBDA-LIST NIL)
  :INLINE T
  :SUBLISTS (2))

(DEF-LIST-FORMAT IL:LAMBDA LAMBDA)

(DEF-LIST-FORMAT LET :INDENT (1)
  :ARGS (:KEYWORD :BINDING-LIST NIL)
  :SUBLISTS (2)
  :MISER :NEVER)

(DEF-LIST-FORMAT LET* LET)

(DEF-LIST-FORMAT LOCALLY :INDENT (0)
  :ARGS (:KEYWORD NIL)
  :SUBLISTS T)

(DEF-LIST-FORMAT LOCALLY :INDENT (0)
  :ARGS (:KEYWORD NIL)
  :SUBLISTS T)

(DEF-LIST-FORMAT LIST :INDENT :HORIZONTAL
  :INLINE T)

(DEF-LIST-FORMAT LIST* LIST)

(DEF-LIST-FORMAT MACROLET FLET)

(DEF-LIST-FORMAT MULTIPLE-VALUE-BIND :INDENT (1 1))
```

```
:ARGS (:KEYWORD :DATA NIL)
:SUBLISTS (2))

(DEF-LIST-FORMAT MULTIPLE-VALUE-SETQ :INDENT (1)
:ARGS (NIL :DATA NIL)
:SUBLISTS (2))

(DEF-LIST-FORMAT IL:NLAMBD LAMBDA)

(DEF-LIST-FORMAT IL:OPENLAMBD LAMBDA)

(DEF-LIST-FORMAT OR AND)

(DEF-LIST-FORMAT IL:P PROGN)

(DEF-LIST-FORMAT PROCEED-CASE :INDENT (1)
:ARGS (NIL :FBINDING))

(DEF-LIST-FORMAT PROG :INDENT (:TAGBODY :STEP 1)
:ARGS (:KEYWORD :BINDING-LIST NIL)
:SUBLISTS (2))

(DEF-LIST-FORMAT PROG* PROG)

(DEF-LIST-FORMAT PROGN :INDENT :VERTICAL
:INLINE NIL)

(DEF-LIST-FORMAT PROGV :INDENT (1 1)
:ARGS (:KEYWORD :DATA NIL)
:SUBLISTS (2))

(DEF-LIST-FORMAT RESTART-CASE :INDENT (1)
:ARGS (NIL :FBINDING))

(DEF-LIST-FORMAT IL:SELCHARQ IL:SELECTQ)

(DEF-LIST-FORMAT IL:SELECTQ :INDENT (1)
:ARGS (:KEYWORD NIL :CASE-CLAUSE)
:LAST NIL)

(DEF-LIST-FORMAT TAGBODY :INDENT (:TAGBODY :STEP 0)
:ARGS (:KEYWORD NIL))

(DEF-LIST-FORMAT THE :INDENT (1)
:ARGS (:KEYWORD NIL)
:INLINE T)

(DEF-LIST-FORMAT TYPECASE CASE)

(DEF-LIST-FORMAT UNDOABLY :INDENT :VERTICAL
:INLINE NIL
:ARGS (:KEYWORD NIL))

(DEF-LIST-FORMAT UNLESS :INDENT (1))

(DEF-LIST-FORMAT UNWIND-PROTECT :INDENT (:BREAK 1))

(DEF-LIST-FORMAT WHEN UNLESS)

(DEF-LIST-FORMAT WITH-INPUT-FROM-STRING WITH-OPEN-STREAM)

(DEF-LIST-FORMAT WITH-OUTPUT-TO-STRING WITH-OPEN-STREAM)

(DEF-LIST-FORMAT WITH-OPEN-FILE WITH-OPEN-STREAM)
```

```
(DEF-LIST-FORMAT WITH-OPEN-STREAM :INDENT (1)
:ARGS (NIL :KEYWORD-ARGS NIL)
:SUBLISTS (2))
```

```
(DEF-LIST-FORMAT IL:WITH.MONITOR :INDENT (1))
```

```
(DEF-LIST-FORMAT IL:WITH.FAST.MONITOR IL:WITH.MONITOR)
```

```
(DEF-LIST-FORMAT IL:WITH.SPY IL:WITH.MONITOR)
```

```
(IL:PUTPROPS IL:SEdit-LIST-FORMATS IL:COPYRIGHT ("Venue & Xerox Corporation" 1987 1988 1990 1992))
```

---

### SEdit-FORMAT INDEX

AND .....	2	DEFOPTIMIZER .....	3	:HORIZONTAL .....	2	PROGV .....	5
:BINDING .....	1	DEFPACKAGE .....	3	:HORIZONTAL-NOBREAK .....	2	RESTART-CASE .....	5
:BINDING-LIST .....	1	DEFPARAMETER .....	3	IF .....	4	RETURN-FROM .....	1
BLOCK .....	1	DEFSETF .....	3	IGNORE-ERRORS .....	4	IL:SELCHARQ .....	5
CASE .....	2	DEFSTRUCT .....	3	:KEYWORD-ARGS .....	2	IL:SELECTQ .....	5
:CASE-CLAUSE .....	2	DEFTYPE .....	3	:KEYWORD-ARGS-NOBREAK .....	2	:SQUASHED .....	2
CATCH .....	2	DEFUN .....	3	LABELS .....	4	TAGBODY .....	5
CCASE .....	2	DEFVAR .....	3	LAMBDA .....	4	THE .....	5
CERROR .....	2	DESTRUCTURING-BIND .....	3	IL:LAMBDA .....	4	THROW .....	1
COMPILER-LET .....	2	DO .....	3	:LAMBDA-LIST .....	1	TYPECASE .....	5
COND .....	2	DO* .....	3	LET .....	4	UNDOABLY .....	5
:COND-CLAUSE .....	2	DO-ALL-SYMBOLS .....	4	LET* .....	4	UNLESS .....	5
CTYPECASE .....	2	DO-EXTERNAL-SYMBOLS .....	4	LIST .....	4	UNWIND-PROTECT .....	5
:DATA .....	1	DO-INTERNAL-SYMBOLS .....	4	LIST* .....	4	:VERTICAL .....	2
DECLARE .....	2	DO-LOCAL-SYMBOLS .....	4	LOCALLY .....	4	:VERTICAL-NOBREAK .....	2
:DEDIT .....	2	DO-SYMBOLS .....	4	MACROLET .....	4	WHEN .....	5
DEF-LIST-FORMAT .....	3	DOLIST .....	3	MULTIPLE-VALUE-BIND .....	4	WITH-INPUT-FROM-STRING .....	5
:DEFAULT .....	1	DOTIMES .....	3	MULTIPLE-VALUE-SETQ .....	5	WITH-OPEN-FILE .....	5
DEFCOMMAND .....	2	ECASE .....	4	IL:NLAMBDA .....	5	WITH-OPEN-STREAM .....	6
DEFCONSTANT .....	3	ERROR .....	4	IL:OPENLAMBDA .....	5	WITH-OUTPUT-TO-STRING .....	5
DEFDEFINER .....	3	ETYPECASE .....	4	OR .....	5	IL:WITH.FAST.MONITOR .....	6
DEFGLOBALPARAMETER .....	3	EVAL-WHEN .....	4	IL:P .....	5	IL:WITH.MONITOR .....	6
DEFGLOBALVAR .....	3	:FBINDING .....	1	PROCEED-CASE .....	5	IL:WITH.SPY .....	6
DEFINE-SPECIAL-FORM .....	3	:FBINDING-LIST .....	2	PROG .....	5		
DEFINLINE .....	3	FLET .....	4	PROG* .....	5		
DEFMACRO .....	3	FORMAT .....	4	PROGN .....	5		

---

### PROPERTY INDEX

IL:SEdit-LIST-FORMATS ...1

---