

File created: 19-Oct-87 14:53:33 {ERINYES}<LISPUSERS>LYRIC>PACKED-STRUCTURE.;1

changes to: (IL:SETFS LOGBITP)  
(IL:FUNCTIONS DEF-PACKED-STRUCTURE SIGNED-LDB)

previous date: 29-Sep-87 18:13:33 {IE:PARC:XEROX}<LISP>LYRIC>LISPUSERS>PACKED-STRUCTURE.;1

Read Table: XCL

Package: XCL-USER

Format: XCCS

; Copyright (c) 1987 by Xerox Corporation. All rights reserved.

```
(IL:RPAQQ IL:PACKED-STRUCTURECOMS ((IL:FUNCTIONS DEF-PACKED-STRUCTURE SIGNED-LDB)
                                     (IL:SETFS LOGBITP)
                                     (IL:PROP IL:MAKEFILE-ENVIRONMENT IL:PACKED-STRUCTURE)))
```

```
(DEFDEFINER DEF-PACKED-STRUCTURE IL:STRUCTURES (NAME &REST SLOTS)
```

```
(LET*
  ((*PACKAGE* (SYMBOL-PACKAGE NAME))
   (COUNT 0)
   (MAX-COUNT 0)
   (LOCATIONS))
  (LABELS
   ((SLOT-NAME (SLOT)
                (CAR SLOT))
    (SLOT-TYPE (SLOT)
                (LET ((TYPE (GETF (CDDR SLOT)
                                   'TYPE T)))
                  (COND
                   ((SUBTYPEP TYPE ' (MEMBER NIL T))
                    ' :BOOLEAN)
                   (T
                     TYPE))))))
   (INFIX (X Y)
           (INTERN (FORMAT NIL "~A--~A" (STRING X)
                                   (STRING Y))))
   (SLOT-LOCATION (SLOT)
                 (CDR (ASSOC (SLOT-NAME SLOT)
                             LOCATIONS)))
   (SLOT-SUPPLIED-P (SLOT)
                    (INFIX (SLOT-NAME SLOT)
                          "SUPPLIED-P"))
   (SLOT-SIGNED (SLOT)
                (AND (LISTP (SLOT-TYPE SLOT))
                     (EQ (CAR (SLOT-TYPE SLOT))
                         'SIGNED-BYTE)))
   (SLOT-SIZE (SLOT)
              (LET ((TYPE (SLOT-TYPE SLOT)))
                (CASE TYPE
                 (:BOOLEAN 1)
                 (T (ECASE (CAR TYPE)
                           ((UNSIGNED-BYTE SIGNED-BYTE) (SECOND TYPE)))))))
   (MAPC #' (LAMBDA (SLOT)
                (WHEN (GETF (CDDR SLOT)
                            ' :OVERLAY)
                  (SETQ COUNT 0))
                (PUSH (CONS (SLOT-NAME SLOT)
                            COUNT)
                      LOCATIONS)
                (INCF COUNT (SLOT-SIZE SLOT))
                (SETQ MAX-COUNT (MAX MAX-COUNT COUNT))))
   SLOTS)
```

; punt for now, this should really coerce other things into stuff  
; that looks like signed or unsigned byte

```
`(PROGN
  (DEFTYPE ,NAME ()
    ' (UNSIGNED-BYTE ,COUNT))
  ,@ (MAPCAR
    #' (LAMBDA (S)
      `(DEFMACRO , (INFIX NAME (SLOT-NAME S)) (X)
        , (COND
          ((EQ (SLOT-TYPE S)
              ' :BOOLEAN)
           `` (LOGBITP ,', (SLOT-LOCATION S)
                     ,X))
          (T `` (', (IF (SLOT-SIGNED S)
                        'SIGNED-LDB
                        'LDB)
                      ,', (BYTE (SLOT-SIZE S)
                                (SLOT-LOCATION S))
                      ,X))))))
    SLOTS)
  (DEFMACRO , (INFIX "MAKE" NAME) (&KEY ,@ (MAPCAR #' (LAMBDA (S)
    (LIST (SLOT-NAME S)
          (SECOND S)
          (SLOT-SUPPLIED-P S))))
```

```

                                SLOTS)
                                &AUX
                                (VALUE 0))
,@ (MAPCAR
  #' (LAMBDA
    (S)
    `(WHEN , (SLOT-SUPPLIED-P S)
      (SETQ VALUE , (COND
        ((EQ (SLOT-TYPE S)
              ',:BOOLEAN)
          `(LOGIOR (IF , , (SLOT-NAME S)
                      ', , (ASH 1 (SLOT-LOCATION S))
                      0)
                  ,VALUE))
        ((SLOT-SIGNED S)
          `(DPB , , (SLOT-NAME S)
                ', , (BYTE (SLOT-SIZE S)
                           (SLOT-LOCATION S))
                ,VALUE))
        (T `(LOGIOR (ASH , , (SLOT-NAME S)
                        ', , (SLOT-LOCATION S))
                  ,VALUE))))))
    SLOTS)
  VALUE))))

```

```

(DEFUN SIGNED-LDB (BYTESPEC INTEGER)
  (FLET ((SIGN-EXTEND (NUMBER POSITION)
    (IF (LOGBITP (1- POSITION)
        NUMBER)
        (DPB NUMBER (BYTE POSITION 0)
              -1)
        NUMBER)))
    (SIGN-EXTEND (LDB BYTESPEC INTEGER)
      (BYTE-SIZE BYTESPEC))))

```

```

(DEFINE-SETF-METHOD LOGBITP (INDEX INTEGER)
  (MULTIPLE-VALUE-BIND (TEMPS VALS STORES STORE-FORM ACCESS-FORM)
    (GET-SETF-METHOD INTEGER) ; get SETF method for integer.
    (LET ((BTEMP (IL:GENSYM)) ; Temp var for index
          (STORE (IL:GENSYM)) ; Temp var for new value
          (STEMP (FIRST STORES)) ; Temp var for int to store.
          )
      (VALUES (CONS BTEMP TEMPS) ; Temporary variables.
              (CONS INDEX VALS) ; Value forms.
              (LIST STORES) ; Store variables.
              `(LET ((,STEMP (IF ,STORE
                                (LOGIOR ,ACCESS-FORM (ASH 1 ,BTEMP))
                                (LOGANDC2 ,ACCESS-FORM (ASH 1 ,BTEMP))))
                    ,STORE-FORM
                    ,STORE) ; Storing form
                `(LOGBITP ,BTEMP ,ACCESS-FORM))))))

```

```

(IL:PUTPROPS IL:PACKED-STRUCTURE IL:MAKEFILE-ENVIRONMENT (:READTABLE "XCL" :PACKAGE "XCL-USER"))

```

```

(IL:PUTPROPS IL:PACKED-STRUCTURE IL:COPYRIGHT ("Xerox Corporation" 1987))

```

---

FUNCTION INDEX

SIGNED-LDB .....2

---

PROPERTY INDEX

IL:PACKED-STRUCTURE .....2

---

SETF INDEX

LOGBITP .....2

---

DEFINER INDEX

DEF-PACKED-STRUCTURE .....1

---