

File created: 16-May-90 13:25:03 {DSK}<usr>local>lde>lispcore>sources>CMLHASH.;2

changes to: (IL:VARS IL:CMLHASHCOMS)

previous date: 8-Jun-89 17:15:50 {DSK}<usr>local>lde>lispcore>sources>CMLHASH.;1

Read Table: XCL

Package: LISP

Format: XCCS

; Copyright (c) 1985, 1986, 1987, 1989, 1990 by Venue & Xerox Corporation. All rights reserved.

(IL:RPAQQ **IL:CMLHASHCOMS**

;; External interface

(IL:FUNCTIONS MAKE-HASH-TABLE GETHASH MAPHASH HASH-TABLE-COUNT HASH-TABLE-P SXHASH)  
(XCL:OPTIMIZERS GETHASH HASH-TABLE-COUNT HASH-TABLE-P)  
(IL:SETFS GETHASH)

;; Internal interface

(IL:FUNCTIONS EQLHASHBITSFN SXHASH-PATHNAME)  
(IL:DECLARE\ : IL:DONTCOPY IL:DOEVAL@COMPILE (IL:VARIABLES SXHASH-MAX)  
(IL:FUNCTIONS SXHASH-LIST SXHASH-STRING SXHASH-BIT-VECTOR SXHASH-ROT))

;; UFN for the SXHASH opcode (a MISCN)

(IL:FNS SXHASH-UFN EQLHASHBITSFN-UFN %SXHASH)  
(XCL:OPTIMIZERS SXHASH EQLHASHBITSFN)  
(XCL:OPTIMIZERS IL:STRINGHASHBITS IL:STRING-EQUAL-HASHBITS)  
(IL:PROP (IL:FILETYPE IL:MAKEFILE-ENVIRONMENT)  
IL:CMLHASH))

;; External interface

(DEFUN **MAKE-HASH-TABLE** (&KEY (TEST 'EQL)  
(SIZE 65)  
REHASH-SIZE REHASH-THRESHOLD)

;; Creates and returns a hash table. See manual for details.

(IF (NOT (SYMBOLP TEST))  
(COND  
((%EQCODEP TEST 'EQ)  
(SETQ TEST 'EQ))  
((%EQCODEP TEST 'EQL)  
(SETQ TEST 'EQL))  
((%EQCODEP TEST 'EQUAL)  
(SETQ TEST 'EQUAL))))  
(ECASE TEST  
(EQ (IL:HASHARRAY SIZE REHASH-SIZE))  
(EQL (IL:HASHARRAY SIZE REHASH-SIZE 'EQLHASHBITSFN 'EQL))  
(EQUAL (IL:HASHARRAY SIZE REHASH-SIZE 'SXHASH 'EQUAL))))

(DEFUN **GETHASH** (KEY HASHTABLE &OPTIONAL DEFAULT)  
(IL:GETHASH KEY HASHTABLE DEFAULT T))

(DEFUN **MAPHASH** (FN HASH-TABLE)  
"Call function with each key/value pair in the hash-table"  
(IL:MAPHASH HASH-TABLE #'(LAMBDA (VALUE KEY)  
(FUNCALL FN KEY VALUE)))  
NIL)

(DEFUN **HASH-TABLE-COUNT** (HASH-TABLE)  
(IL:HARRAYPROP HASH-TABLE 'IL:NUMKEYS))

(DEFUN **HASH-TABLE-P** (OBJECT)  
(IL:TYPENAMEP OBJECT 'IL:HARRAYP))

(DEFUN **SXHASH** (OBJECT)  
(IL:MISCN SXHASH OBJECT))

(XCL:DEFOPTIMIZER **GETHASH** (KEY HASHTABLE &OPTIONAL DEFAULT XCL:&CONTEXT CONTEXT)  
(IF (EQ 1 (COMPILER:CONTEXT-VALUES-USED CONTEXT))  
(IF DEFAULT  
'(IL:GETHASH ,KEY ,HASHTABLE ,DEFAULT)  
'(IL:GETHASH ,KEY ,HASHTABLE))  
'COMPILER:PASS))

(XCL:DEFOPTIMIZER **HASH-TABLE-COUNT** (HASH-TABLE)

```
`(IL:HARRAYPROP ,HASH-TABLE 'IL:NUMKEYS))
```

```
(XCL:DEFOPTIMIZER HASH-TABLE-P (OBJECT)
` (IL:TYPENAMEP ,OBJECT 'IL:HARRAYP))
```

```
(DEFSETF GETHASH PUTHASH)
```

:: Internal interface

```
(DEFUN EQLHASHBITSFN (OBJ)
(IL:MISCN EQLHASHBITSFN OBJ)
```

```
(DEFUN SXHASH-PATHNAME (PATHNAME)
(LET ((HASH (SXHASH-ROT (LOGXOR (%SXHASH (IL:%PATHNAME-HOST PATHNAME))
(%SXHASH (IL:%PATHNAME-DEVICE PATHNAME))))))
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (%SXHASH (IL:%PATHNAME-TYPE PATHNAME))))))
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (%SXHASH (IL:%PATHNAME-VERSION PATHNAME))))))
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (%SXHASH (IL:%PATHNAME-DIRECTORY PATHNAME))))))
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (%SXHASH (IL:%PATHNAME-NAME PATHNAME))))))
```

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

```
(DEFCONSTANT SXHASH-MAX 13)
```

```
(DEFMACRO SXHASH-LIST (LIST)
` (DO ((LIST ,LIST (CDR LIST))
(INDEX 0 (1+ INDEX))
(HASH 0))
(OR (NOT (CONSP LIST))
(EQ INDEX SXHASH-MAX))
HASH)
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (%SXHASH (CAR LIST))))))
```

```
(DEFMACRO SXHASH-STRING (STRING) ; Returns hash value for a general string.
```

```
` (DO ((I 0 (1+ I))
(LENGTH (MIN (LENGTH ,STRING)
SXHASH-MAX))
(HASH 0))
(EQ I LENGTH)
HASH)
```

:: the spice code had a fairly general "rotate X within integerlength of most-positive-fixnum bits, but (a) it was slow and (b) it was buggy anyway, since it assumed that most-positive-fixnum was 1 less than a power of two.

```
(SETQ HASH (SXHASH-ROT (LOGXOR HASH (CHAR-INT (AREF ,STRING I))))))
```

```
(DEFMACRO SXHASH-BIT-VECTOR (BIT-VECTOR)
```

```
` (DO ((I 0 (1+ I))
(LENGTH (MIN (LENGTH ,BIT-VECTOR)
16))
(HASH 0))
(EQ I LENGTH)
HASH)
(SETQ HASH (+ (ASH HASH 1)
(AREF ,BIT-VECTOR I))))
```

```
(DEFMACRO SXHASH-ROT (X)
```

```
` (LET ((X ,X))
(DPB X (BYTE 9 7)
(LDB (BYTE 7 9)
X)))
)
```

:: UFN for the SXHASH opcode (a MISCN)

```
(IL:DEFINEQ
```

```
(SXHASH-UFN
```

```
(IL:LAMBDA (IL:INDEX IL:ARGCOUNT IL:ARG-PTR) ; Edited 23-Feb-89 19:45 by jds
```

:: This is the UFN for the CL: SXHASH MISCN sub-opcode. That MISCN is being implemented on Suns.

```
(%SXHASH (IL:\GETBASEPTR IL:ARG-PTR 0))))
```

```
(EQLHASHBITSFN-UFN
```

```
(IL:LAMBDA (IL:INDEX IL:ARGCOUNT IL:ARG-PTR) ; Edited 23-Feb-89 18:10 by jds
```

```
(LET ((OBJ (IL:\GETBASEPTR IL:ARG-PTR 0))
(TYPECASE OBJ
```

```

(CHARACTER (CHAR-INT OBJ))
(INTEGER (LOGAND OBJ 65535))
(FLOAT (LOGXOR (IL:|fetch| (IL:FLOATP IL:HIWORD) IL:|of| OBJ)
              (IL:|fetch| (IL:FLOATP IL:LOWORD) IL:|of| OBJ)))
(RATIO (LOGXOR (EQLHASHBITSFN (NUMERATOR OBJ))
              (EQLHASHBITSFN (DENOMINATOR OBJ))))
(COMPLEX (LOGXOR (EQLHASHBITSFN (REALPART OBJ))
                (EQLHASHBITSFN (IMAGPART OBJ))))
(T (IL:\EQHASHINGBITS OBJ))))

```

**(%SXHASH**

```

(IL:LAMBDA (OBJECT) ; Edited 23-Feb-89 19:42 by jds
(COND
((SYMBOLP OBJECT)
 (IL:\EQHASHINGBITS OBJECT))
(LISTP OBJECT)
(SXHASH-LIST OBJECT))
(NUMBERP OBJECT)
(TYPECASE OBJECT
 (INTEGER (LOGAND OBJECT MOST-POSITIVE-FIXNUM))
 (FLOAT (LOGXOR (IL:|fetch| (IL:FLOATP IL:HIWORD) IL:|of| OBJECT)
                (IL:|fetch| (IL:FLOATP IL:LOWORD) IL:|of| OBJECT)))
 (RATIO (LOGXOR (%SXHASH (NUMERATOR OBJECT))
                (%SXHASH (DENOMINATOR OBJECT))))
 (COMPLEX (LOGXOR (%SXHASH (REALPART OBJECT))
                  (%SXHASH (IMAGPART OBJECT)))))
((STRINGP OBJECT)
(SXHASH-STRING OBJECT))
(BIT-VECTOR-P OBJECT)
(SXHASH-BIT-VECTOR OBJECT))
(PATHNAMEP OBJECT)
(SXHASH-PATHNAME OBJECT))
(T (IL:\EQHASHINGBITS OBJECT))))
)

```

```

(XCL:DEFOPTIMIZER SXHASH (OBJECT)
                  `(IL:MISCN SXHASH ,OBJECT))

```

```

(XCL:DEFOPTIMIZER EQLHASHBITSFN (OBJECT)
                  `(IL:MISCN EQLHASHBITSFN ,OBJECT))

```

```

(XCL:DEFOPTIMIZER IL:STRINGHASHBITS (STRING)
                  `(IL:MISCN IL:STRINGHASHBITS ,STRING))

```

```

(XCL:DEFOPTIMIZER IL:STRING-EQUAL-HASHBITS (STRING)
                  `(IL:MISCN IL:STRING-EQUAL-HASHBITS ,STRING))

```

```

(IL:PUTPROPS IL:CMLHASH IL:FILETYPE COMPILER-FILE)

```

```

(IL:PUTPROPS IL:CMLHASH IL:MAKEFILE-ENVIRONMENT (:READTABLE "XCL" :PACKAGE "LISP"))

```

```

(IL:PUTPROPS IL:CMLHASH IL:COPYRIGHT ("Venue & Xerox Corporation" 1985 1986 1987 1989 1990))

```

---

**FUNCTION INDEX**

%SXHASH .....	3	GETHASH .....	1	MAKE-HASH-TABLE .....	1	SXHASH-PATHNAME .....	2
EQLHASHBITSFN .....	2	HASH-TABLE-COUNT .....	1	MAPHASH .....	1	SXHASH-UFN .....	2
EQLHASHBITSFN-UFN .....	2	HASH-TABLE-P .....	1	SXHASH .....	1		

---

**OPTIMIZER INDEX**

EQLHASHBITSFN .....	3	HASH-TABLE-COUNT .....	1	IL:STRING-EQUAL-HASHBITS	3	SXHASH .....	3
GETHASH .....	1	HASH-TABLE-P .....	2	IL:STRINGHASHBITS .....	3		

---

**MACRO INDEX**

SXHASH-BIT-VECTOR .....	2	SXHASH-LIST .....	2	SXHASH-ROT .....	2	SXHASH-STRING .....	2
-------------------------	---	-------------------	---	------------------	---	---------------------	---

---

**PROPERTY INDEX**

IL:CMLHASH .....

---

**SETF INDEX**

GETHASH .....

---

**CONSTANT INDEX**

SXHASH-MAX .....

---