

File created: 22-May-91 09:08:00 {DSK}<new>sources>lispcore>sources>TIME.;2

previous date: 17-May-90 15:51:58 {DSK}<new>sources>lispcore>sources>TIME.;1

Read Table: XCL

Package: LISP

Format: XCCS

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

```
(IL:RPAQQ IL:TIMECOMS
  ((IL:STRUCTURES STATS-OBJECT)
   (IL:FUNCTIONS %COPY-TIME-STATS %STATS-OBJECT-DIFFERENCE)
   (IL:FUNCTIONS %GET-TIMING-INFO TIME-CALL TIME)
   (IL:FUNCTIONS %CAPTURE-COUNTERS-BEFORE %CAPTURE-COUNTERS-AFTER TIME-FORMAT %PRINT-TIMING-ITEM
    %PRINT-TIMING-INFO)
   (IL:DECLARE\ : IL:DONTCOPY IL:DOEVAL@COMPILE (IL:FUNCTIONS %CAPTURE-BEFORE-STATS %CAPTURE-AFTER-STATS
    %MOVE-FIXP-FIELD))
   (IL:SPECIAL-FORMS TIME)
   (IL:COMMANDS "TIME")
   ;; Interlisp Timeall function
   (IL:FNS IL:TIMEALL)
   ;; file package stuff
   (IL:PROP IL:FILETYPE TIME)
   (IL:PROP IL:MAKEFILE-ENVIRONMENT TIME)
   (IL:DECLARE\ : IL:DONTEVAL@LOAD IL:DOEVAL@COMPILE IL:DONTCOPY (IL:LOCALVARS . T))
   (IL:DECLARE\ : IL:DONTEVAL@LOAD IL:DOEVAL@COMPILE IL:DONTCOPY IL:COMPILERVARS (IL:ADDVARS (IL:NLAMA
    (IL:NLAML
     IL:TIMEALL
    )
   (IL:LAMA))))))

(DEFSTRUCT (STATS-OBJECT (:TYPE LIST)
  (:COPIER NIL)
  (:PREDICATE NIL))
  (ELAPSED-TIME (IL:CLOCK 0))
  (TIME-BLOCK (IL:|create| IL:MISCSTATS))
  (DATA-COUNTERS (MAKE-ARRAY (1+ IL:|\MaxTypeNumber|)
    :ELEMENT-TYPE
    '(SIGNED-BYTE 32)
    :INITIAL-ELEMENT 0))
  DATATYPES)

(DEFUN %COPY-TIME-STATS (REFERENCE-BLOCK DESTINATION-BLOCK)
  ;; Copies various fields from one miscstats block to another. Both reference-block and destination-block should be unboxed hunks (made by
  ;; (IL:create IL:miscstats)), but IL:\miscstats is also a valid value for reference-block
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:SWAPWAITTIME)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:GCTIME)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:PAGEFAULTS)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:SWAPWRITES)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:TOTALTIME)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:DISKIOTIME)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:NETIOTIME)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  (%MOVE-FIXP-FIELD (IL:MISCSTATS IL:DISKOPS)
    DESTINATION-BLOCK REFERENCE-BLOCK)
  DESTINATION-BLOCK)

(DEFUN %STATS-OBJECT-DIFFERENCE (BEFORE AFTER)
  ;; puts the differences between the stat-object after and stat-object before back into after.
  (LET ((BEFORE-DATA-COUNTERS (STATS-OBJECT-DATA-COUNTERS BEFORE))
        (BEFORE-TIME-BLOCK (STATS-OBJECT-TIME-BLOCK BEFORE))
        (AFTER-DATA-COUNTERS (STATS-OBJECT-DATA-COUNTERS AFTER))
        (AFTER-TIME-BLOCK (STATS-OBJECT-TIME-BLOCK AFTER)))
    (DOTIMES (I (LENGTH BEFORE-DATA-COUNTERS))
      (DECF (AREF AFTER-DATA-COUNTERS I)
        (AREF BEFORE-DATA-COUNTERS I)))
    (DECF (STATS-OBJECT-ELAPSED-TIME AFTER)
      (STATS-OBJECT-ELAPSED-TIME BEFORE))
    (DECF (IL:|fetch| (IL:MISCSTATS IL:SWAPWAITTIME) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:SWAPWAITTIME) IL:|of| BEFORE-TIME-BLOCK))
    (DECF (IL:|fetch| (IL:MISCSTATS IL:GCTIME) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:GCTIME) IL:|of| BEFORE-TIME-BLOCK))
```

```
(DECF (IL:|fetch| (IL:MISCSTATS IL:DISKIOTIME) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:DISKIOTIME) IL:|of| BEFORE-TIME-BLOCK))
(DECF (IL:|fetch| (IL:MISCSTATS IL:PAGEFAULTS) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:PAGEFAULTS) IL:|of| BEFORE-TIME-BLOCK))
(DECF (IL:|fetch| (IL:MISCSTATS IL:SWAPWRITES) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:SWAPWRITES) IL:|of| BEFORE-TIME-BLOCK))
(DECF (IL:|fetch| (IL:MISCSTATS IL:DISKOPS) IL:|of| AFTER-TIME-BLOCK)
      (IL:|fetch| (IL:MISCSTATS IL:DISKOPS) IL:|of| BEFORE-TIME-BLOCK))
AFTER))
```

```
(DEFUN %GET-TIMING-INFO (TIMED-FUNCTION TIME-BEFORE TIME-AFTER &OPTIONAL (REPEAT 1))
  ;; Side-effects TIME-BEFORE and TIME-AFTER. Returns the value (or values of TIMED-FUNCTION, and the timing-info in TIME-AFTER.
```

```
(LET ((VALUES NIL))
  (%CAPTURE-BEFORE-STATS TIME-BEFORE)
  (DOTIMES (I (1- REPEAT))
    (FUNCALL TIMED-FUNCTION))
  (SETQ VALUES (MULTIPLE-VALUE-LIST (FUNCALL TIMED-FUNCTION)))
  (%CAPTURE-AFTER-STATS TIME-AFTER)
  (%STATS-OBJECT-DIFFERENCE TIME-BEFORE TIME-AFTER)
  (VALUES-LIST VALUES)))
```

```
(DEFUN TIME-CALL (TIMED-FUNCTION &KEY (OUTPUT *TRACE-OUTPUT*)
  (TIMED-FORM NIL TIMED-FORM-P)
  (DATA-TYPES (IL:DATATYPES))
  (REPEAT 1))
```

```
(LET ((VALUES NIL)
      (TIME-BEFORE (MAKE-STATS-OBJECT))
      (TIME-AFTER (MAKE-STATS-OBJECT))
      (TIME-DO-NOTHING (MAKE-STATS-OBJECT)))
  ;; Calibrate
  (%GET-TIMING-INFO #'(LAMBDA NIL NIL)
    TIME-BEFORE TIME-DO-NOTHING)
  (SETQ VALUES (MULTIPLE-VALUE-LIST (%GET-TIMING-INFO TIMED-FUNCTION TIME-BEFORE TIME-AFTER REPEAT)))
  (%STATS-OBJECT-DIFFERENCE TIME-DO-NOTHING TIME-AFTER)
  (IF TIMED-FORM-P (TIME-FORMAT OUTPUT "Timing for ~[~;~:~:* ~D x~]:~20T ~S~&" REPEAT TIMED-FORM))
  (%PRINT-TIMING-ITEM OUTPUT "Elapsed time" (STATS-OBJECT-ELAPSED-TIME TIME-AFTER)
    T T)
  (%PRINT-TIMING-INFO OUTPUT TIME-AFTER DATA-TYPES)
  (VALUES-LIST VALUES)))
```

```
(DEFMACRO TIME (TIMED-FORM &REST KEYWORDS)
  `(TIME-CALL #'(LAMBDA NIL ,TIMED-FORM)
    :TIMED-FORM
    ',TIMED-FORM
    ,@KEYWORDS))
```

```
(DEFUN %CAPTURE-COUNTERS-BEFORE (VECTOR)
```

;; Record box count for all known datatypes before timing. Note, IL:BOXCOUNT may create fixp's, so count down, so the FIXP count is recorded last

```
(DO ((I (1- (LENGTH VECTOR))
      (1- I)))
  ((< I 0)
   VECTOR)
  (SETF (AREF VECTOR I)
    (IL:BOXCOUNT I))))
```

```
(DEFUN %CAPTURE-COUNTERS-AFTER (VECTOR)
```

;; Record box count for all known datatypes after timing. Note, IL:BOXCOUNT may create fixp's, so count up, so the FIXP count is recorded first

```
(DOTIMES (I (LENGTH VECTOR)
  VECTOR)
  (SETF (AREF VECTOR I)
    (IL:BOXCOUNT I))))
```

```
(DEFUN TIME-FORMAT (STREAM FORMAT-STRING &REST ARGS)
```

```
(IF (EQ STREAM :EXEC)
  (APPLY 'XCL:EXEC-FORMAT FORMAT-STRING ARGS)
  (APPLY 'FORMAT STREAM FORMAT-STRING ARGS)))
```

```
(DEFUN %PRINT-TIMING-ITEM (STREAM STRING NUM TIME-P ALWAYS-P)
```

```
(IF (OR ALWAYS-P (> NUM 0))
  (IF TIME-P
    (TIME-FORMAT STREAM "~&~A ~20,5T= ~9,3F seconds~&" STRING (MAX 0 (/ NUM 1000.0)))
    (TIME-FORMAT STREAM "~&~A ~20,5T= ~9D~&" STRING NUM))))
```

```
(DEFUN %PRINT-TIMING-INFO (STREAM STATS-OBJECT DATA-TYPES)
```

```
(LET ((TIME-BLOCK (STATS-OBJECT-TIME-BLOCK STATS-OBJECT)))
```

```
(DATA-TYPE-INFO (LET ((DATA-COUNTER (STATS-OBJECT-DATA-COUNTERS STATS-OBJECT))
  (RESULT NIL)
  (RESULT-TAIL NIL)
  CNT TYPE-NAME)
  (DOTIMES (I (MIN (LENGTH DATA-COUNTER)
    (1+ IL:|\\MaxTypeNumber|))
    RESULT)
    (SETQ CNT (AREF DATA-COUNTER I))
    (WHEN (> CNT 0)
      (SETQ TYPE-NAME (IL:\\TYPENAMEFROMNUMBER I))
      (IF (MEMBER TYPE-NAME DATA-TYPES :TEST #'EQ)
        (IF RESULT
          (RPLACD RESULT-TAIL (SETQ RESULT-TAIL (LIST (LIST CNT TYPE-NAME)))
            )
          (SETQ RESULT (SETQ RESULT-TAIL (LIST (LIST CNT TYPE-NAME))))))))))
)
(%PRINT-TIMING-ITEM STREAM "SWAP time" (IL:|fetch| (IL:MISCSTATS IL:SWAPWAITTIME) IL:|of| TIME-BLOCK)
T)
(%PRINT-TIMING-ITEM STREAM "reclaim time" (IL:|fetch| (IL:MISCSTATS IL:GCTIME) IL:|of| TIME-BLOCK)
T)
(%PRINT-TIMING-ITEM STREAM "Disk i/o time" (IL:|fetch| (IL:MISCSTATS IL:DISKIOTIME) IL:|of| TIME-BLOCK)
T)
(%PRINT-TIMING-ITEM STREAM "net compute time" (- (STATS-OBJECT-ELAPSED-TIME STATS-OBJECT)
  (IL:|fetch| (IL:MISCSTATS IL:SWAPWAITTIME) IL:|of| TIME-BLOCK)
  )
  (IL:|fetch| (IL:MISCSTATS IL:GCTIME) IL:|of| TIME-BLOCK)
  (IL:|fetch| (IL:MISCSTATS IL:DISKIOTIME) IL:|of| TIME-BLOCK)
  (IL:|fetch| (IL:MISCSTATS IL:NETIOTIME) IL:|of| TIME-BLOCK)
  )
T)
(%PRINT-TIMING-ITEM STREAM "Page faults" (IL:|fetch| (IL:MISCSTATS IL:PAGEFAULTS) IL:|of| TIME-BLOCK))
(%PRINT-TIMING-ITEM STREAM "Swap writes" (IL:|fetch| (IL:MISCSTATS IL:SWAPWRITES) IL:|of| TIME-BLOCK))
(%PRINT-TIMING-ITEM STREAM "Disk operations" (IL:|fetch| (IL:MISCSTATS IL:DISKOPS) IL:|of| TIME-BLOCK))
(IF DATA-TYPE-INFO (TIME-FORMAT STREAM "~&Storage allocated:~%~{~D ~A~}~^, ~}~&" DATA-TYPE-INFO)
(TIME-FORMAT STREAM "~%"))
```

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

```
(DEFMACRO %CAPTURE-BEFORE-STATS (STATS-OBJECT)
;; Capture machine state before timing an evaluation. Note that ordering is important
` (LET ((%$$STATS-OBJECT ,STATS-OBJECT))
  (%CAPTURE-COUNTERS-BEFORE (STATS-OBJECT-DATA-COUNTERS %$$STATS-OBJECT))
  (%COPY-TIME-STATS IL:\\MISCSTATS (STATS-OBJECT-TIME-BLOCK %$$STATS-OBJECT))
  (IL:CLOCK0 (STATS-OBJECT-ELAPSED-TIME %$$STATS-OBJECT))))
```

```
(DEFMACRO %CAPTURE-AFTER-STATS (STATS-OBJECT)
` (LET ((%$$STATS-OBJECT ,STATS-OBJECT))
  (IL:CLOCK0 (STATS-OBJECT-ELAPSED-TIME %$$STATS-OBJECT))
  (%COPY-TIME-STATS IL:\\MISCSTATS (STATS-OBJECT-TIME-BLOCK %$$STATS-OBJECT))
  (%CAPTURE-COUNTERS-AFTER (STATS-OBJECT-DATA-COUNTERS %$$STATS-OBJECT))))
```

```
(DEFMACRO %MOVE-FIXP-FIELD (FIELD-NAME DEST SOURCE)
` (IL:\\BLT (IL:LOC F (IL:FETCH ,FIELD-NAME IL:OF ,DEST))
  (IL:LOC F (IL:FETCH ,FIELD-NAME IL:OF ,SOURCE))
  2))
)
```

```
(XCL:DEFINE-SPECIAL-FORM TIME (TIMED-FORM &KEY (DATA-TYPES ' (IL:DATATYPES))
  (REPEAT 1)
  (OUTPUT '*TRACE-OUTPUT*
    &ENVIRONMENT ENV &AUX *EVALHOOK* *APPLYHOOK*))
(TIME-CALL #' (LAMBDA NIL (EVAL TIMED-FORM ENV))
  :TIMED-FORM TIMED-FORM :DATA-TYPES (EVAL DATA-TYPES ENV)
  :REPEAT
  (EVAL REPEAT ENV)
  :OUTPUT
  (EVAL OUTPUT ENV)))
```

```
(XCL:DEFCOMMAND "TIME" (FORM &KEY (REPEAT 1)
  &ENVIRONMENT ENV) "Time evaluation of form, output here"
(TIME-CALL #' (LAMBDA NIL (EVAL FORM ENV))
  :OUTPUT :EXEC :REPEAT (EVAL REPEAT ENV)))
```

;; Interlisp Timeall function

```
(IL:DEFINEQ
```

```
(IL:TIMEALL
```

```
(IL:NLAMBDA (IL:TIMEFORM IL:NUMBEROFTIMES IL:TIMEWHAT IL:INTERPFLG)
```

; Edited 29-Jan-87 18:48 by jop

;; collects and prints stats on TIMEFORM. TIMEWHAT indicates what to collect stats on: if T, all of the system times are collected; if NIL, the system times plus all data allocations are kept; if a list, it should be a list of DATATYPES (or numbers) .

```

(LET ((IL:DATATYPES (COND
  ((NULL IL:TIMEWHAT)
   (IL:DATATYPES))
  ((EQ IL:TIMEWHAT T)
   NIL)
  (T (IL:|for| IL:X IL:|inside| IL:TIMEWHAT IL:|bind| IL:NAME
      IL:|join| (COND
        ((IL:SETQ IL:NAME (IL:DATATYPEP IL:X))
         (CONS IL:NAME))
        ((EQ IL:X 'TIME)
         NIL)
        (T (IL:|printout| T IL:X " is not a datatype." T)
         NIL))))))
  IL:VALUE)
 (OR (IL:NUMBERP IL:NUMBEROFTIMES)
      (IL:SETQ IL:NUMBEROFTIMES 1))
 (LET ((IL:STRF T)
       (IL:LCFIL NIL))
  (DECLARE (IL:SPECVARS IL:STRF IL:LCFIL)
            (IL:COMPILE1 'IL:TIMEDUMMYFUNCTION `(IL:LAMBDA NIL
                                                , IL:TIMEFORM))
            (TIME-CALL 'IL:TIMEDUMMYFUNCTION :OUTPUT (IL:GETSTREAM NIL 'IL:OUTPUT)
                       :TIMED-FORM IL:TIMEFORM :DATA-TYPES IL:DATATYPES :REPEAT IL:NUMBEROFTIMES))))))
)

;; file package stuff

(IL:PUTPROPS TIME IL:FILETYPE COMPILE-FILE)
(IL:PUTPROPS TIME IL:MAKEFILE-ENVIRONMENT (:READTABLE "XCL" :PACKAGE "LISP"))
(IL:DECLARE\ : IL:DONTEVAL@LOAD IL:DOEVAL@COMPILE IL:DONTCOPY
(IL:DECLARE\ : IL:DOEVAL@COMPILE IL:DONTCOPY
(IL:LOCALVARS . T)
)
)
(IL:DECLARE\ : IL:DONTEVAL@LOAD IL:DOEVAL@COMPILE IL:DONTCOPY IL:COMPILERVERS
(IL:ADDTOVAR IL:NLAMA )
(IL:ADDTOVAR IL:NLAML IL:TIMEALL)
(IL:ADDTOVAR IL:LAMA )
)
(IL:PUTPROPS TIME IL:COPYRIGHT ("Venue & Xerox Corporation" 1986 1987 1988 1990 1991))

```

FUNCTION INDEX

%CAPTURE-COUNTERS-AFTER .2 %GET-TIMING-INFO2 %STATS-OBJECT-DIFFERENCE 1 IL:TIMEALL3
%CAPTURE-COUNTERS-BEFORE 2 %PRINT-TIMING-INFO2 TIME-CALL2
%COPY-TIME-STATS1 %PRINT-TIMING-ITEM2 TIME-FORMAT2

MACRO INDEX

%CAPTURE-AFTER-STATS3 %CAPTURE-BEFORE-STATS ...3 %MOVE-FIXP-FIELD3 TIME2

PROPERTY INDEX

TIME4

COMMAND INDEX

"TIME"3

SPECIAL-FORM INDEX

TIME3

STRUCTURE INDEX

STATS-OBJECT1
