

File created: 15-Sep-87 11:28:32 {MCS:MCS:STANFORD}<LANE>TCPTIME.;22

changes to: (VARS TCPTIMECOMS)

previous date: 14-Sep-87 08:59:11 {MCS:MCS:STANFORD}<LANE>TCPTIME.;21

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

::
:: Copyright (c) 1986, 1987 by Stanford University. All rights reserved.

(RPAQQ TCPTIMECOMS

```
(( * * Common TCP and UDP Time Client and Server Functions)
 (FNS RFC868.SETTIME RFC868.START.SERVER RFC868.STOP.SERVER)
 (INITVARS (RFC868.TIME.PORT 37)
            (RFC868.DEFAULT.PROTOCOL 'TCP))
 (ADDVARS (RFC868.ASCII.OSTYPES VMS))
 (* Constant adjusts Jan 1, 1901 by one year (in seconds)
    since lisp will not accept Jan 1, 1900)
 [DECLARE%: DONTCOPY (CONSTANTS (RFC868.START.OF.TIME (DIFFERENCE (TIMES 365 24 60 60)
                                                                    (IDATE " 1-Jan-01 00:00:00 GMT"))
                                (RFC868.DEFAULT.PROTOCOL RFC868.ASCII.OSTYPES RFC868.PROTOCOLS))
 (GLOBALVARS RFC868.TIME.PORT RFC868.DEFAULT.PROTOCOL RFC868.ASCII.OSTYPES RFC868.PROTOCOLS)
 (FNS RFC868.IDATE RFC868.SETNEWTIME)
 (* * TCP Time Client and Server)
 (FNS TCP.SETTIME TCP.TIMESERVER)
 (ADDVARS (RFC868.PROTOCOLS (TCP TCP.SETTIME TCP.TIMESERVER)))
 (INITVARS TCP.TIME.HOSTS (TCP.TIME.PORT RFC868.TIME.PORT)
            (TCP.SETTIME.TIMEOUT 10000))
 (GLOBALVARS TCP.TIME.HOSTS TCP.TIME.PORT TCP.SETTIME.TIMEOUT)
 (DECLARE%: DONTCOPY (MACROS READTIME WRITETIME))
 (FILES TCP)
 (* * UDP Time Client and Server)
 (FNS UDP.SETTIME UDP.TIMESERVER)
 (ADDVARS (RFC868.PROTOCOLS (UDP UDP.SETTIME UDP.TIMESERVER)))
 (INITVARS UDP.TIME.HOSTS (UDP.TIME.PORT RFC868.TIME.PORT)
            (UDP.SETTIME.TIMEOUT 10000))
 (GLOBALVARS UDP.TIME.HOSTS UDP.TIME.PORT UDP.SETTIME.TIMEOUT)
 (DECLARE%: DONTCOPY (MACROS UDP.APPEND.TIME GETBASETIME))
 (FILES TCPUDP)))
```

(* * Common TCP and UDP Time Client and Server Functions)

(DEFINEQ

(RFC868.SETTIME

[LAMBDA (RETFLG PROTOCOL)

; Edited 10-Sep-87 11:03 by cdl
(* DECLARATIONS%: (RECORD SERVICE
(PROTOCOL CLIENT SERVER)))

```
(LET (SERVICE)
  (if (SETQ SERVICE (ASSOC (OR PROTOCOL RFC868.DEFAULT.PROTOCOL)
                           RFC868.PROTOCOLS))
    then (with SERVICE SERVICE (APPLY* CLIENT RETFLG]))
```

(RFC868.START.SERVER

[LAMBDA (PROTOCOL ASCIIFLG)

; Edited 10-Sep-87 11:03 by cdl
(* DECLARATIONS%: (RECORD SERVICE
(PROTOCOL CLIENT SERVER)))

```
(LET (SERVICE)
  (if [AND (SETQ SERVICE (ASSOC (OR PROTOCOL RFC868.DEFAULT.PROTOCOL)
                                RFC868.PROTOCOLS))
        (with SERVICE SERVICE (NOT (FIND.PROCESS SERVER))
        then (with SERVICE SERVICE (ADD.PROCESS `(:,SERVER ,ASCIIFLG)
                                              'RESTARTABLE T]))
```

(RFC868.STOP.SERVER

[LAMBDA (PROTOCOL)

; Edited 10-Sep-87 11:03 by cdl
(* DECLARATIONS%: (RECORD SERVICE
(PROTOCOL CLIENT SERVER)))

```
(LET (SERVICE PROCESS)
  (if [AND (SETQ SERVICE (ASSOC (OR PROTOCOL RFC868.DEFAULT.PROTOCOL)
                                RFC868.PROTOCOLS))
        (with SERVICE SERVICE (SETQ PROCESS (FIND.PROCESS SERVER))
        then (DEL.PROCESS PROCESS]))
```

)

(RPAQ? RFC868.TIME.PORT 37)

(RPAQ? RFC868.DEFAULT.PROTOCOL 'TCP)


```

{MEDLEY}<lispusers>TCPTIME.;1

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS TCP.TIME.HOSTS TCP.TIME.PORT TCP.SETTIME.TIMEOUT)
)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(PUTPROPS READTIME MACRO ((STREAM)
                          (PLUS (LSH (BIN16 STREAM)
                                     16)
                                (BIN16 STREAM))))

(PUTPROPS WRITETIME MACRO [(STREAM TIME)
                          (BOUT16 STREAM (RSH TIME 16))
                          (BOUT16 STREAM (LOGAND TIME (MASK.1'S 0 16]))
)
)

(FILESLoad TCP)

(* * UDP Time Client and Server)

(DEFINEQ

(UDP.SETTIME
  [LAMBDA (RETFLG)
    (DECLARE (SPECVARS RETFLG))
    (LET (SOCKET)
      (DECLARE (SPECVARS SOCKET))
      (RESETLST
        [RESETSAVE NIL `(UDP.CLOSE.SOCKET , (SETQ SOCKET (UDP.OPEN.SOCKET)
        (bind UDP ADDRESS RESULT for HOST in UDP.TIME.HOSTS
          when (AND (SETQ ADDRESS (DODIP.HOSTP HOST))
                    (SETQ UDP (UDP.EXCHANGE SOCKET (UDP.SETUP (\ALLOCATE.ETHERPACKET)
                                                                ADDRESS UDP.TIME.PORT 0 SOCKET 'FREE)
                                                                UDP.SETTIME.TIMEOUT))
                    (SETQ RESULT (RFC868.SETNEWTIME (with UDP UDP (GETBASETIME UDPCONTENTS 0))
                                                                RETFLG)))]
        do (RETURN RESULT)))]])
; Edited 10-Sep-87 13:20 by cdl

(UDP.TIMESERVER
  [LAMBDA NIL
    (LET (SOCKET)
      (DECLARE (SPECVARS SOCKET))
      (RESETLST
        [RESETSAVE NIL `(UDP.CLOSE.SOCKET , (SETQ SOCKET (UDP.OPEN.SOCKET UDP.TIME.PORT)
        (bind UDP while (SETQ UDP (UDP.GET SOCKET T)) do (UDP.SETUP UDP (with IP UDP IPSOURCEADDRESS)
          (with UDP UDP UDPSOURCEPORT)
            0 SOCKET 'FREE)
          (UDP.APPEND.TIME UDP (RFC868.IDATE))
          (UDP.SEND SOCKET UDP)))]])
; Edited 10-Sep-87 13:04 by cdl

)

(ADDTovar RFC868.PROTOCOLS (UDP UDP.SETTIME UDP.TIMESERVER))

(RPAQ? UDP.TIME.HOSTS NIL)

(RPAQ? UDP.TIME.PORT RFC868.TIME.PORT)

(RPAQ? UDP.SETTIME.TIMEOUT 10000)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS UDP.TIME.HOSTS UDP.TIME.PORT UDP.SETTIME.TIMEOUT)
)

(DECLARE%: DONTCOPY

(DECLARE%: EVAL@COMPILE

(PUTPROPS UDP.APPEND.TIME MACRO [OPENLAMBDA (UDP TIME)
                          (UDP.APPEND.WORD UDP (RSH TIME 16))
                          (UDP.APPEND.WORD UDP (LOGAND TIME (MASK.1'S 0 16]))

(PUTPROPS GETBASETIME MACRO [(OFFSET BASE)
                          (PLUS (LSH (\GETBASE OFFSET BASE)
                                     16)
                                (\GETBASE OFFSET (ADD1 BASE))]
)
)

(FILESLoad TCPUDP)

```

```
{MEDLEY}<lispusers>TCPTIME.;1
```

Page 4

```
(PUTPROPS TCPTIME COPYRIGHT ("Stanford University" 1986 1987))
```

FUNCTION INDEX

RFC868.IDATE	2	RFC868.START.SERVER	1	TCP.TIMESERVER	2
RFC868.SETNEWTIME	2	RFC868.STOP.SERVER	1	UDP.SETTIME	3
RFC868.SETTIME	1	TCP.SETTIME	2	UDP.TIMESERVER	3

VARIABLE INDEX

RFC868.ASCII.OSTYPES	2	RFC868.TIME.PORT	1	TCP.TIME.PORT	2	UDP.TIME.PORT	3
RFC868.DEFAULT.PROTOCOL .	1	TCP.SETTIME.TIMEOUT	2	UDP.SETTIME.TIMEOUT	3		
RFC868.PROTOCOLS	2,3	TCP.TIME.HOSTS	2	UDP.TIME.HOSTS	3		

MACRO INDEX

GETBASETIME	3	READTIME	3	UDP.APPEND.TIME	3	WRITETIME	3
-------------------	---	----------------	---	-----------------------	---	-----------------	---

CONSTANT INDEX

RFC868.START.OF.TIME2
