

File created: 23-May-88 18:55:42 {ERIS}<MULLINS>RPC>CURRENT>RPCOS.;11

changes to: (IL:VARS IL:RPCOSCOMS)

previous date: 20-May-88 12:56:30 {ERIS}<MULLINS>RPC>CURRENT>RPCOS.;9

Read Table: XCL

Package: RPC2

Format: XCCS

; Copyright (c) 1988 by Xerox Corporation. All rights reserved.

```
(IL:RPAQQ IL:RPCOSCOMS
  ( ;; OS networking code
    (IL:PROPS (IL:RPCOS IL:MAKEFILE-ENVIRONMENT IL:FILETYPE))
    (EVAL-WHEN (COMPILE)
      (IL:FILES (IL:LOADCOMP)
        IL:LLSUBRS))
    (IL:FUNCTIONS OS-EXCHANGE-UDP-PACKETS OS-RESOLVE-HOST READ-STRING-ADDRESS)
  ;; XDR data block
    (IL:RECORDS XDR-DATA-BLOCK)
    (IL:VARIABLES *CELLS-PER-XDR-DATA-BLOCK* *FREE-XDR-DATA-BLOCKS* *MAX-XDR-DATA-BLOCKS*
      *WORDS-PER-CELL*)
    (IL:FUNCTIONS XDR-INITIALIZE-CACHE ALLOCATE-XDR-DATA-BLOCK RECLAIM-XDR-DATA-BLOCK)
    (IL:FUNCTIONS FOLDLO UNFOLD)
    (IL:FUNCTIONS OS-UDP-GETBYTE OS-UDP-GETBYTES OS-UDP-PUTBYTE OS-UDP-PUTBYTES OS-UDP-GETCELL
      OS-UDP-PUTCELL OS-UDP-GETOFFSET OS-UDP-PUTOFFSET)
    (EVAL-WHEN (LOAD)
      (IL:P (XDR-INITIALIZE-CACHE))))))

;; OS networking code
(IL:PUTPROPS IL:RPCOS IL:MAKEFILE-ENVIRONMENT (:READTABLE "XCL" :PACKAGE "RPC2"))
(IL:PUTPROPS IL:RPCOS IL:FILETYPE :COMPILE-FILE)
(EVAL-WHEN (COMPILE)
  (IL:FILESLOAD (IL:LOADCOMP)
    IL:LLSUBRS)
)

(DEFUN OS-EXCHANGE-UDP-PACKETS (RPCSTREAM MSEC-UNTIL-TIMEOUT MSEC-BETWEEN-TRIES ERRORFLG)
  (LET ((XDR-ARG-BLOCK (RPC-STREAM-OUTSTREAM RPCSTREAM))
        (XDR-ARG-BLOCK-LENGTH (RPC-STREAM-OUTBYTEPTR RPCSTREAM))
        (XDR-RESULT-BLOCK (RPC-STREAM-INSTREAM RPCSTREAM)))
    (SETQ XDR-ARG-BLOCK (IL:\DTEST XDR-ARG-BLOCK 'XDR-DATA-BLOCK))
    (SETQ XDR-RESULT-BLOCK (IL:\DTEST XDR-RESULT-BLOCK 'XDR-DATA-BLOCK))
  ;; Need to add a dispatch on the type of the error.
    (UNLESS (IL:SUBRCALL IL:RPC-CALL (RPC-STREAM-OS-DESTADDR RPCSTREAM)
      (RPC-STREAM-IPSOCKET RPCSTREAM)
      XDR-ARG-BLOCK XDR-RESULT-BLOCK MSEC-UNTIL-TIMEOUT MSEC-BETWEEN-TRIES XDR-ARG-BLOCK-LENGTH
    )
      (CASE ERRORFLG
        (:NOERRORS (THROW 'GOFORIT NIL))
        (:RETURNERRORS (THROW 'GOFORIT ' (ERROR TIMEOUT)))
        (OTHERWISE (ERROR "RPC Call failed"))))
  ;; Put the result block in the instream.
    (SETF (RPC-STREAM-INSTREAM RPCSTREAM)
      XDR-RESULT-BLOCK)
    (PROGN (WHEN *DEBUG*
      (FORMAT-T "It returned!~%"
        (AND (NUMBERP *DEBUG*)
          (> *DEBUG* 5)
          (BREAK "Reply Packet in INSTREAM of RPC-STREAM *RPCSTREAM*"))
      T)))

(DEFUN OS-RESOLVE-HOST (DESTINATION)
  ;; Convert an address from it's string representation into a number.
  (LET ((ADDR (READ-STRING-ADDRESS (IF (SYMBOLP DESTINATION)
    DESTINATION
    (INTERN DESTINATION "IL")))))
    (IF ADDR
      ADDR
      DESTINATION)))
```


(DEFGLOBALPARAMETER ***FREE-XDR-DATA-BLOCKS*** NIL
"A list of free xdr data blocks.")

(DEFGLOBALPARAMETER ***MAX-XDR-DATA-BLOCKS*** 5
"The maximum size of the data block cache.")

(DEFCONSTANT ***WORDS-PER-CELL*** 2
"The number of words (16 bits) per cell.")

(DEFUN **XDR-INITIALIZE-CACHE** ()
(LET ((CACHE-LENGTH (LENGTH *FREE-XDR-DATA-BLOCKS*))
(UNLESS (>= CACHE-LENGTH *MAX-XDR-DATA-BLOCKS*)
(DOTIMES (I (- *MAX-XDR-DATA-BLOCKS* CACHE-LENGTH))
(PUSH (IL:**create**| XDR-DATA-BLOCK)
FREE-XDR-DATA-BLOCKS))))))

(DEFUN **ALLOCATE-XDR-DATA-BLOCK** ()
"If an xdr data block is available then return it, otherwise create one."
(OR (POP *FREE-XDR-DATA-BLOCKS*)
(IL:**create**| XDR-DATA-BLOCK)))

(DEFUN **RECLAIM-XDR-DATA-BLOCK** (XDR-DATA-BLOCK)
(WHEN (< (LENGTH *FREE-XDR-DATA-BLOCKS*)
MAX-XDR-DATA-BLOCKS)
(PUSH XDR-DATA-BLOCK *FREE-XDR-DATA-BLOCKS*))
T)

(DEFMACRO **FOLDLO** (FORM DIVISOR)
(LET ((DIV (IF (CONSTANTP DIVISOR)
(EVAL DIVISOR)
DIVISOR)))
(OR (AND DIV (IL:POWEROFTWOP DIV))
(IL:\\ILLEGAL.ARG DIV))
(LIST 'IL:LRSH FORM (IL:SUB1 (IL:INTEGERLENGTH DIV))))))

(DEFMACRO **UNFOLD** (FORM DIVISOR)
(LET ((DIV (IF (CONSTANTP DIVISOR)
(EVAL DIVISOR)
DIVISOR)))
(OR (AND DIV (IL:POWEROFTWOP DIV))
(IL:\\ILLEGAL.ARG DIV))
(LIST 'IL:LLSH FORM (IL:SUB1 (IL:INTEGERLENGTH DIV))))))

(DEFUN **OS-UDP-GETBYTE** (RPCSTREAM)
"Get a byte from the instream of the rpcstream and increment the offset."
(LET ((OFFSET (RPC-STREAM-INBYTEPTR RPCSTREAM))
(XDR-DATA-BLOCK (RPC-STREAM-INSTREAM RPCSTREAM)))
(PROG1 (IL:\\GETBASEBYTE (IL:LOCF (IL:**fetch**| XDR-PUBLIC IL:**of**| (SETQ XDR-DATA-BLOCK (IL:\\DTEST
XDR-DATA-BLOCK
'XDR-DATA-BLOCK))
))
OFFSET)
(SETF (RPC-STREAM-INBYTEPTR RPCSTREAM)
(+ 1 OFFSET))))))

(DEFUN **OS-UDP-GETBYTES** (RPCSTREAM NBYTES)
"Get nbytes bytes from the rpcstream and increment the offset"
(LET* ((XDR-DATA-BLOCK (RPC-STREAM-INSTREAM RPCSTREAM))
(STRING (IL:ALLOCSTRING NBYTES)))
(IL:\\MOVEBYTES (IL:LOCF (IL:**fetch**| (XDR-DATA-BLOCK XDR-PUBLIC) IL:**of**| (SETQ XDR-DATA-BLOCK
(IL:\\DTEST XDR-DATA-BLOCK
'XDR-DATA-BLOCK)))))
(OR (IL:SMALLP (RPC-STREAM-INBYTEPTR RPCSTREAM))
0)
(IL:**fetch**| (IL:STRINGP IL:BASE) IL:**of**| STRING)
(IL:**fetch**| (IL:STRINGP IL:OFFST) IL:**of**| STRING)
NBYTES)
(INCF (RPC-STREAM-INBYTEPTR RPCSTREAM)
NBYTES)
STRING))

(DEFUN **OS-UDP-PUTBYTE** (RPCSTREAM BYTE)
"Put a byte of data at the next position in the rpcstream and increment the offset."
(LET ((XDR-DATA-BLOCK (RPC-STREAM-OUTSTREAM RPCSTREAM))
(IL:\\PUTBASEBYTE (IL:LOCF (IL:**fetch**| XDR-PUBLIC IL:**of**| (SETQ XDR-DATA-BLOCK (IL:\\DTEST XDR-DATA-BLOCK
'XDR-DATA-BLOCK)))))
(RPC-STREAM-OUTBYTEPTR RPCSTREAM)

```

      BYTE))
      (INCF (RPC-STREAM-OUTBYTEPTR RPCSTREAM)
            1))

```

```

(DEFUN OS-UDP-PUTBYTES (RPCSTREAM STRING)
  "Put a string of bytes into the outstream of rpcstream and increment the offset."
  (LET ((XDR-DATA-BLOCK (RPC-STREAM-OUTSTREAM RPCSTREAM))
        (LENGTH (LENGTH STRING)))
    (IL:\MOVEBYTES (IL:|fetch| (IL:STRINGP IL:BASE) IL:|of| STRING)
                   (IL:|fetch| (IL:STRINGP IL:OFFST) IL:|of| STRING)
                   (IL:LOCF (IL:|fetch| (XDR-DATA-BLOCK XDR-PUBLIC) IL:|of| (SETQ XDR-DATA-BLOCK (IL:\DTEST
                                                                 XDR-DATA-BLOCK
                                                                 'XDR-DATA-BLOCK))))
                   )
    (RPC-STREAM-OUTBYTEPTR RPCSTREAM)
    LENGTH)
  (INCF (RPC-STREAM-OUTBYTEPTR RPCSTREAM)
        LENGTH)))

```

```

(DEFUN OS-UDP-GETCELL (RPCSTREAM)
  "Get a cell from the rpcstream and increment the offset."
  (LET* ((BYTEOFFSET (RPC-STREAM-INBYTEPTR RPCSTREAM))
         (XDR-DATA-BLOCK (RPC-STREAM-INSTREAM RPCSTREAM))
         (CELL# (ASH BYTEOFFSET -2)))
    (IF (AND (>= CELL# 0)
             (<= CELL# *CELLS-PER-XDR-DATA-BLOCK*))
        (LET ((BASE (IL:LOCF (IL:|fetch| (XDR-DATA-BLOCK XDR-PUBLIC) IL:|of| (SETQ XDR-DATA-BLOCK
                                                                 (IL:\DTEST XDR-DATA-BLOCK
                                                                 'XDR-DATA-BLOCK))))))
          (PROG1 (IL:\MAKENUMBER (IL:\GETBASE BASE (UNFOLD CELL# *WORDS-PER-CELL*))
                                (IL:\GETBASE BASE (1+ (UNFOLD CELL# *WORDS-PER-CELL*))))
                (SETF (RPC-STREAM-INBYTEPTR RPCSTREAM)
                      (+ 4 BYTEOFFSET))))
          (ERROR "Attempt to fetch cell outside of buffer.")))
        (ERROR "Attempt to fetch cell outside of buffer.")))

```

```

(DEFUN OS-UDP-PUTCELL (RPCSTREAM VALUE)
  (LET* ((BYTEOFFSET (RPC-STREAM-OUTBYTEPTR RPCSTREAM))
         (XDR-DATA-BLOCK (RPC-STREAM-OUTSTREAM RPCSTREAM))
         (BASE (IL:LOCF (IL:|fetch| (XDR-DATA-BLOCK XDR-PUBLIC) IL:|of| (SETQ XDR-DATA-BLOCK (IL:\DTEST
                                                                 XDR-DATA-BLOCK
                                                                 'XDR-DATA-BLOCK))))
         )))
  (IL:\PUTBASEBYTE BASE BYTEOFFSET (LDB (BYTE 8 24)
                                         VALUE))
  (IL:\PUTBASEBYTE BASE (IL:\ADDBASE BYTEOFFSET 1)
                    (LDB (BYTE 8 16)
                          VALUE))
  (IL:\PUTBASEBYTE BASE (IL:\ADDBASE BYTEOFFSET 2)
                    (LDB (BYTE 8 8)
                          VALUE))
  (IL:\PUTBASEBYTE BASE (IL:\ADDBASE BYTEOFFSET 3)
                    (LDB (BYTE 8 0)
                          VALUE))
  (SETF (RPC-STREAM-OUTBYTEPTR RPCSTREAM)
        (+ 4 BYTEOFFSET)))

```

```

(DEFUN OS-UDP-GETOFFSET (RPCSTREAM)
  (CONS (RPC-STREAM-INSTREAM RPCSTREAM)
        (RPC-STREAM-INBYTEPTR RPCSTREAM)))

```

```

(DEFUN OS-UDP-PUTOFFSET (RPCSTREAM BYTEOFFSET)
  (SETF (RPC-STREAM-INBYTEPTR RPCSTREAM)
        BYTEOFFSET))

```

```

(EVAL-WHEN (LOAD)

```

```

(XDR-INITIALIZE-CACHE)
)

```

```

(IL:PUTPROPS IL:RPCOS IL:COPYRIGHT ("Xerox Corporation" 1988))

```

FUNCTION INDEX

ALLOCATE-XDR-DATA-BLOCK .3 OS-UDP-GETBYTES3 OS-UDP-PUTBYTES4 RECLAIM-XDR-DATA-BLOCK ..3
OS-EXCHANGE-UDP-PACKETS .1 OS-UDP-GETCELL4 OS-UDP-PUTCELL4 XDR-INITIALIZE-CACHE3
OS-RESOLVE-HOST1 OS-UDP-GETOFFSET4 OS-UDP-PUTOFFSET4
OS-UDP-GETBYTE3 OS-UDP-PUTBYTE3 READ-STRING-ADDRESS2

MACRO INDEX

FOLDLO3 UNFOLD3

CONSTANT INDEX

CELLS-PER-XDR-DATA-BLOCK2 *WORDS-PER-CELL*3

VARIABLE INDEX

FREE-XDR-DATA-BLOCKS3 *MAX-XDR-DATA-BLOCKS*3

PROPERTY INDEX

IL:RPCOS1

RECORD INDEX

XDR-DATA-BLOCK2
