

File created: 19-Mar-2024 18:24:39 {WMEDLEY}<sources>EXTERNALFORMAT.;88

edit by: rmk

changes to: (FNS \FORMATBYTESTRING \FORMATBYTESTREAM)

previous date: 12-Jan-2024 10:59:18 {WMEDLEY}<sources>EXTERNALFORMAT.;86

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

(RPAQQ EXTERNALFORMATCOMS

```
[[COMS ; EXTERNALFORMAT declaration and related functions
; (originally on FILEIO)
(DECLARE%: DOEVAL@COMPILE DONTCOPY (EXPORT (RECORDS EXTERNALFORMAT)))
(INITRECORDS EXTERNALFORMAT)
(SYSRECORDS EXTERNALFORMAT)
(FNS \EXTERNALFORMAT MAKE-EXTERNALFORMAT \EXTERNALFORMAT.DEFFPRINT)
(FNS \INSTALL.EXTERNALFORMAT \REMOVE.EXTERNALFORMAT FIND-FORMAT)
(FNS SYSTEM-EXTERNALFORMAT)
(GLOBALVARS *EXTERNALFORMATS* *DEFAULT-EXTERNALFORMAT*)
(INITVARS (*EXTERNALFORMATS* NIL)
(*DEFAULT-EXTERNALFORMAT* ;XCCS))
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (DEFFPRINT 'EXTERNALFORMAT (FUNCTION \EXTERNALFORMAT.DEFFPRINT)
[COMS ;; Generic functions not compiled open (originally on LLREAD)
(FNS \OUTCHAR \INCCODE \BACKCCODE \BACKCCODE.EOLC \PEEKCCODE \PEEKCCODE.EOLC \INCCODE.EOLC
\FORMATBYTESTREAM \FORMATBYTESTRING \CHECKEOLC.CRLF)
(DECLARE%: DOEVAL@COMPILE DONTCOPY (EXPORT (MACROS \CHECKEOLC))
(RESOURCES \FORMATBYTESTRING.STREAM))
(INITRESOURCES \FORMATBYTESTRING.STREAM))
[COMS ; NULL device, from FILEIO
(FNS \NULLDEVICE \NULL.OPENFILE)
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (\NULLDEVICE))
[COMS ;; Also from FILEIO, but not clear that this is or ever has been used.
(FNS \CREATE.THROUGH.EXTERNALFORMAT \THROUGHIN \THROUGHBACKCCODE \THROUGHOUTCHARFN)
(DECLARE%: DONTEVAL@LOAD DOCOPY (P (\CREATE.THROUGH.EXTERNALFORMAT]))
```

;; EXTERNALFORMAT declaration and related functions (originally on FILEIO)

```
(DECLARE%: DOEVAL@COMPILE DONTCOPY
```

;; FOLLOWING DEFINITIONS EXPORTED

```
(DECLARE%: EVAL@COMPILE
```

```
(DATATYPE EXTERNALFORMAT ((EOLVALID FLAG) ; If true, the value of EOL field will replace the
; EOLCONVENTION field of the resulted stream. (Can't test EOL
; because it is always something)
(EOL BITS 2) ; T if (like XCCS runcodes) the byte encoding of a given
(UNSTABLE FLAG) ; character can change by other signals in the file, NIL if every
; charactercode has a single byte encoding (like UTF-8).
(INCCODEFN POINTER) ; Called with STREAM and 2 optional arguments,
; BYTECOUNTVAR and BYTECOUNTVAL
(PEEKCCODEFN POINTER) ; Called with three arguments -- STREAM, NOERROR, and EOL
(BACKCCODEFN POINTER) ; Called with STREAM and optional BYTECOUNTVAR and
; BYTECOUNTVAL
(OUTCHARFN POINTER) ; Called with two arguments -- STREAM and CHARCODE
(NAME POINTER) ; keyword name of this format, provided to
; \INSTALL.EXTERNALFORMAT
(FORMATBYTESTREAMFN POINTER) ; Function to copy the format state of a given stream to an IO
; stream that allows formatted byte sequences to be examined
(EF1 POINTER) ; Extra fields for use of particular formats. Possibly to hold
; standardized translation tables
(EF2 POINTER)
(FORMATBYTESTRINGFN POINTER) ; Translates an internal string into a string containing the bytes
; that represent that string in this format
(FORMATCHARSETFN POINTER) ; If present, apply by \GENERIC.CHARSET
```

```
)
```

```
(/DECLAREDATATYPE 'EXTERNALFORMAT ' (FLAG (BITS 2)
FLAG POINTER POINTER POINTER POINTER POINTER POINTER POINTER
POINTER POINTER)
```

;; ---field descriptor list elided by lister---

```
' 20)
```

```
)
```

;; END EXPORTED DEFINITIONS

```
(/DECLAREDATATYPE 'EXTERNALFORMAT' (FLAG (BITS 2)
FLAG POINTER POINTER POINTER POINTER POINTER POINTER POINTER
POINTER POINTER))
```

```
;; ---field descriptor list elided by lister---
' 20)
```

```
(ADDTOVAR SYSTEMRECLST
(DATATYPE EXTERNALFORMAT ((EOLVALID FLAG)
(EOL BITS 2)
(UNSTABLE FLAG)
(INCCODEFN POINTER)
(PEEKCCODEFN POINTER)
(BACKCCODEFN POINTER)
(OUTCHARFN POINTER)
(NAME POINTER)
(FORMATBYTESTREAMFN POINTER)
(EF1 POINTER)
(EF2 POINTER)
(FORMATBYTESTRINGFN POINTER)
(FORMATCHARSETFN POINTER))))
```

```
(DEFINEQ
```

(EXTERNALFORMAT

```
[LAMBDA (STREAM NEWFORMAT/NAME)
;; Edited 2-Jul-2022 19:17 by rmk: Fast case: NEWFORMAT/NAME is an external format
;; Edited 22-Jun-2022 09:40 by rmk: NEWFORMAT/NAME can be a stream, picks its externalformat
;; Edited 10-Sep-2021 20:44 by rmk:
;; Edited 26-Feb-91 13:20 by nm

;;; July 2020: Added interface for per-device default external format. \DO.PARAMS.AT.OPEN will make that call even if it is not specified from the
;;; open. STREAMPROP is fixed to call \EXTERNALFORMAT to set the property EXTERNALFORMAT, to export a user-level way of manipulating this.
;;;
;;; If NEWFORMAT/NAME is nil, just returns the current external format name of STREAM. If NEWFORMAT/NAME is supplied and it is or names an
;;; external format, then the external format of STREAM is set to that format.
;;;
;;; :DEFAULT means the default external format for STREAM's filedevice. If a different format is not specified when the device is created, it will default
;;; to the value of *DEFAULT-EXTERNALFORMAT*, initialized in FILEIO.
```

```
(\DTEST STREAM 'STREAM)
(CL:WHEN NEWFORMAT/NAME
[LET ((EXTFORMAT NEWFORMAT/NAME)
(CL:UNLESS (type? EXTERNALFORMAT EXTFORMAT)
;; Try for coercions
(CL:WHEN (type? READER-ENVIRONMENT NEWFORMAT/NAME)
(SETQ NEWFORMAT/NAME (fetch (READER-ENVIRONMENT REFORMAT) of NEWFORMAT/NAME)))
(if (type? EXTERNALFORMAT NEWFORMAT/NAME)
then (SETQ EXTFORMAT NEWFORMAT/NAME)
elseif (\GETSTREAM NEWFORMAT/NAME NIL T)
then (SETQ EXTFORMAT (ffetch (STREAM EXTERNALFORMAT) of NEWFORMAT/NAME))
else (CL:WHEN (EQ NEWFORMAT/NAME :DEFAULT)
(SETQ NEWFORMAT/NAME (fetch (FDEV DEFAULTEXTERNALFORMAT) of (fetch (STREAM DEVICE)
of STREAM))))
(SETQ EXTFORMAT (FIND-FORMAT NEWFORMAT/NAME NIL STREAM))
(CL:UNLESS EXTFORMAT (ERROR NEWFORMAT/NAME "is not a registered external format name"))
(CL:UNLESS (type? EXTERNALFORMAT EXTFORMAT)
(ERROR "INVALID EXTERNALFORMAT " EXTFORMAT)))
(UNINTERRUPTABLY
(freplace (STREAM EXTERNALFORMAT) of STREAM with EXTFORMAT)
(CL:WHEN (ffetch (EXTERNALFORMAT EOLVALID) of EXTFORMAT)
(freplace (STREAM EOLCONVENTION) of STREAM with (ffetch (EXTERNALFORMAT EOL) of EXTFORMAT)))
(freplace (STREAM OUTCHARFN) of STREAM with (ffetch (EXTERNALFORMAT OUTCHARFN) of EXTFORMAT))
(freplace (STREAM INCCODEFN) of STREAM with (ffetch (EXTERNALFORMAT INCCODEFN) of EXTFORMAT))
(freplace (STREAM PEEKCCODEFN) of STREAM with (ffetch (EXTERNALFORMAT PEEKCCODEFN) of EXTFORMAT))
(freplace (STREAM BACKCCODEFN) of STREAM with (ffetch (EXTERNALFORMAT BACKCCODEFN) of EXTFORMAT))))
(fetch (EXTERNALFORMAT NAME) of (fetch (STREAM EXTERNALFORMAT) of STREAM))
```

(MAKE-EXTERNALFORMAT

```
[LAMBDA (NAME INCCODEFN PEEKCCODEFN BACKCCODEFN OUTCHARFN FORMATBYTESTREAMFN EOL UNSTABLE FORMATBYTESTRINGFN
DEFAULT FORMATCHARSETFN)
; Edited 8-Dec-2023 22:02 by rmk
; Edited 3-Jul-2022 00:35 by rmk
; Edited 10-Sep-2021 19:47 by rmk:

;; Compiled creator for EXTERNALFORMAT so that declaration (EXPORTS.ALL) is not needed. If EOL is not specified, then EOLVALID is also
;; NIL. Fills in missing functions from DEFAULT if given. If DEFAULT is T, use *DEFAULT-EXTERNALFORMAT*.

(CL:WHEN DEFAULT
[LET [(DEF (FIND-FORMAT (CL:IF (EQ DEFAULT T)
```

DEFAULT-EXTERNALFORMAT
DEFAULT)]

(CL:UNLESS INCCODEFN
(SETQ INCCODEFN (FETCH (EXTERNALFORMAT INCCODEFN)
DEF)))
(CL:UNLESS PEEKCCODEFN
(SETQ PEEKCCODEFN (FETCH (EXTERNALFORMAT PEEKCCODEFN)
DEF)))
(CL:UNLESS BACKCCODEFN
(SETQ BACKCCODEFN (FETCH (EXTERNALFORMAT BACKCCODEFN)
DEF)))
(CL:UNLESS OUTCHARFN
(SETQ OUTCHARFN (FETCH (EXTERNALFORMAT OUTCHARFN)
DEF))))]

(SETQ EOL (SELECTC EOL
(LIST 'LF LF.EOLC)
LF.EOLC)
(LIST 'CR CR.EOLC)
CR.EOLC)
(LIST 'CRLF CRLF.EOLC)
CRLF.EOLC)
(NIL)
(SHOULDNT)))
(INSTALL.EXTERNALFORMAT (CREATE EXTERNALFORMAT
NAME _ NAME
INCCODEFN _ INCCODEFN
PEEKCCODEFN _ PEEKCCODEFN
BACKCCODEFN _ BACKCCODEFN
OUTCHARFN _ OUTCHARFN
FORMATBYTESTREAMFN _ FORMATBYTESTREAMFN
EOLVALID _ EOL
EOL _ (OR EOL LF.EOLC)
UNSTABLE _ UNSTABLE
FORMATBYTESTRINGFN _ FORMATBYTESTRINGFN
FORMATCHARSETFN _ (OR FORMATCHARSETFN (FUNCTION NIL]))

(EXTERNALFORMAT.DEFPRINT

[LAMBDA (EXTERNALFORMAT STREAM) ; Edited 2-Jul-2022 11:40 by rmk
; Edited 8-May-87 15:55 by bvm

;; Print device using its name, for example, #<EXTERNALFORMAT:UTF-8/76,5432>
(\DEFPRINT.BY.NAME EXTERNALFORMAT STREAM (fetch (EXTERNALFORMAT NAME) of EXTERNALFORMAT)
"EXTERNALFORMAT"])

)

(DEFINEQ

(INSTALL.EXTERNALFORMAT

[LAMBDA (EXTFORMAT/NAME EXTERNALFORMAT) ; Edited 5-Aug-2021 14:22 by rmk:

;;; Register an instance of the datatype EXTERNALFORMAT.

;;; For backward compatibility, the first argument can be a NAME with the second argument being the format. If so, the NAME must match the name
;;; inside the format

(LET (NAME)
(IF EXTERNALFORMAT
THEN ;; Backwards compatibility
(SETQ NAME (MKATOM EXTFORMAT/NAME))
(IF (EQ NAME (FETCH (EXTERNALFORMAT NAME) OF EXTERNALFORMAT))
ELSEIF (FETCH (EXTERNALFORMAT NAME) OF EXTERNALFORMAT)
THEN (ERROR "Mismatch of specified name and name of the external format")
ELSE (REPLACE (EXTERNALFORMAT NAME) OF EXTERNALFORMAT WITH NAME))
ELSE (SETQ EXTERNALFORMAT EXTFORMAT/NAME)
(SETQ NAME (FETCH (EXTERNALFORMAT NAME) OF EXTERNALFORMAT)))
(IF (type? EXTERNALFORMAT EXTERNALFORMAT)
THEN (\REMOVE.EXTERNALFORMAT NAME)
(push *EXTERNALFORMATS* EXTERNALFORMAT)
ELSE (ERROR "INVALID EXTERNALFORMAT " EXTERNALFORMAT))
EXTERNALFORMAT])

(REMOVE.EXTERNALFORMAT

[LAMBDA (NAME/EXTFORMAT) ; Edited 5-May-2021 15:42 by rmk:

;;; Deregisters external format EXTERNALFORMAT .

(SETQ NAME/EXTFORMAT (IF (TYPE? EXTERNALFORMAT NAME/EXTFORMAT)
THEN (FETCH (EXTERNALFORMAT NAME) OF NAME/EXTFORMAT)
ELSE (MKATOM NAME/EXTFORMAT)))
(SETQ *EXTERNALFORMATS* (DREMOVE (FIND EF IN *EXTERNALFORMATS* SUCHTHAT (EQ NAME/EXTFORMAT
(FETCH (EXTERNALFORMAT NAME)
OF EF)))
EXTERNALFORMATS))

(FIND-FORMAT

```
[LAMBDA (NAME NOERROR) ; Edited 2-Jul-2022 18:55 by rmk
; Edited 7-Aug-2021 09:29 by rmk:
  (IF (TYPE? EXTERNALFORMAT NAME)
      THEN NAME
      ELSE (SETQ NAME (MKATOM NAME))) ; The EQMEMB allows for synonyms, the first of which should be
; canonical. E.g. (:UTF-8 :UTF8)
  (OR (FIND EF IN *EXTERNALFORMATS* SUCHTHAT (EQ NAME (FETCH (EXTERNALFORMAT NAME) OF EF)))
      (CL:UNLESS NOERROR (ERROR NAME "is not an external format"))))
```

)

(DEFINEQ

(SYSTEM-EXTERNALFORMAT

```
[LAMBDA NIL ; Edited 10-Oct-2022 11:55 by Imm
; Edited 7-Jul-2022 10:41 by rmk
  (FOR X IN ' ("LC_CTYPE" "LC_ALL" "LANG") WHEN (STRPOS ".UTF-8" (UNIX-GETENV X)) DO (RETURN :UTF-8)
  FINALLY (RETURN :THROUGH))
```

)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS *EXTERNALFORMATS* *DEFAULT-EXTERNALFORMAT*)

)

(RPAQ? *EXTERNALFORMATS* NIL)

(RPAQ? *DEFAULT-EXTERNALFORMAT* :XCCS)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(DEFPRINT 'EXTERNALFORMAT (FUNCTION \EXTERNALFORMAT.DEFPRINT))

)

:: Generic functions not compiled open (originally on LLREAD)

(DEFINEQ

(\OUTCHAR

```
[LAMBDA (STREAM CODE) ; Edited 30-Jun-2022 10:02 by rmk
; Edited 10-Aug-2021 10:29 by rmk:
  ;; We can't do the EOL stuff here because we don't know whether BOUTs are legit.
  ;; Maybe the implementation function does something else, like move the X and Y positions. At best we could convert the EOL into either CR or
  ;; LF, or into a CR-LF sequence that we pass by two calls to the lower implementation function.
  ;;
  ;; This would make CHARPOSITION generic:
```

```
(* (FREPLACE (STREAM CHARPOSITION) OF STREAM
WITH (CL:IF (EQ CODE (CHARCODE EOL)) 0
(IPLUS16 1 (FFETCH (STREAM CHARPOSITION) OF
STREAM))))))
```

```
(CL:FUNCALL (ffetch (STREAM OUTCHARFN) of STREAM)
STREAM CODE)
CODE])
```

(\INCCODE

```
[LAMBDA (STREAM BYTECOUNTVAR BYTECOUNTVAL) ; Edited 30-Jun-2022 10:04 by rmk
; Edited 7-Aug-2021 00:11 by rmk:
```

:: Calling functions pass the name of the BYTECOUNTVAR, or NIL. If non-NIL, implementing functions are required to SETQ *BYTECOUNTER* to the number of bytes read (positive) or backed up (negative).

:: Caller must bind BYTECOUNTVAR as a SPECVAR. BYTECOUNTVAL can be passed as the current value of BYTECOUNTVAR, to save a call to \EVALV1.

```
(IF BYTECOUNTVAR
  THEN [LET ((*BYTECOUNTER* 0))
        (DECLARE (SPECVARS *BYTECOUNTER*))
        (PROG1 (CL:FUNCALL (ffetch (STREAM INCCODEFN) of STREAM)
                          STREAM
                          '*BYTECOUNTER*)
              (SET BYTECOUNTVAR (IDIFFERENCE (OR BYTECOUNTVAL (\EVALV1 BYTECOUNTVAR))
                                             *BYTECOUNTER*)))]
  ELSE (CL:FUNCALL (ffetch (STREAM INCCODEFN) of STREAM)
                STREAM])
```

(\BACKCODE

```
[LAMBDA (STREAM BYTECOUNTVAR BYTECOUNTVAL) ; Edited 12-Jan-2024 10:58 by Imm
; Edited 19-Jul-2022 15:55 by rmk
; Edited 30-Jun-2022 10:00 by rmk
; Edited 14-Aug-2021 00:26 by rmk:
```

:: Format function returns the backed-over character code if the backup succeed, NIL otherwise (e.g at the beginning of the file). FIXP test and
:: PEEKCCODE here for implementations that don't want to bother computing the code from the bytes.

```
(LET (CODE)
  (IF BYTECOUNTVAR
    THEN [LET ((*BYTECOUNTER* 0))
      (DECLARE (SPECVARS *BYTECOUNTER*))
      (SETQ CODE (CL:FUNCALL (ffetch (STREAM BACKCCODEFN) of STREAM)
        STREAM T))
      (SET BYTECOUNTVAR (IPLUS (OR BYTECOUNTVAL (\EVALV1 BYTECOUNTVAR))
        *BYTECOUNTER*))
      (CL:WHEN CODE
        (OR (FIXP CODE)
          (CL:FUNCALL (ffetch (STREAM PEEKCCODEFN) of STREAM)
            STREAM)))]
    ELSEIF (SETQ CODE (CL:FUNCALL (ffetch (STREAM BACKCCODEFN) of STREAM)
      STREAM))
    THEN (OR (FIXP CODE)
      (CL:FUNCALL (ffetch (STREAM PEEKCCODEFN) of STREAM)
        STREAM]))
```

(\BACKCCODE.EOLC

```
[LAMBDA (STREAM EOLC BYTECOUNTVAR BYTECOUNTVAL) ; Edited 20-Jul-2022 13:05 by rmk
; Edited 30-Jun-2022 10:02 by rmk
; Edited 18-Jun-2022 18:45 by rmk
; Edited 14-Aug-2021 00:27 by rmk:
```

:: If the EOLCONVENTION is CRLF, and the first backup is over an LF encoding, this looks to see whether the preceding bytes encode a CR and if
:: so, backs up over those.

```
(CL:UNLESS EOLC
  (SETQ EOLC (ffetch (STREAM EOLCONVENTION) OF STREAM)))
(LET (CODE (*BYTECOUNTER* 0))
  (DECLARE (SPECVARS *BYTECOUNTER*))
  :: In almost all cases, we just execute the first backup
  (SELCHARQ (SETQ CODE (CL:FUNCALL (ffetch (STREAM BACKCCODEFN) of STREAM)
    STREAM BYTECOUNTVAR))
    (LF (SELECTC EOLC
      ((LIST LF.EOLC 'LF)
        (SETQ CODE (CHARCODE EOL)))
      ((LIST CRLF.EOLC ANY.EOLC 'CRLF 'ANY)
        (SETQ CODE (CHARCODE EOL)) ; Also an EOL, but back over a preceding CR
        (CL:UNLESS (EQ (CHARCODE CR)
          (CL:FUNCALL (ffetch (STREAM BACKCCODEFN) of STREAM)
            STREAM BYTECOUNTVAR))
          :: Not a preceding CR, reread it.
          (CL:FUNCALL (ffetch (STREAM INCCODEFN) of STREAM)
            STREAM BYTECOUNTVAR)))
        NIL))
      (CR (CL:WHEN [MEMB EOLC (CONSTANT (LIST CR.EOLC ANY.EOLC 'CR 'ANY])
        (SETQ CODE (CHARCODE EOL))))
        NIL)
      (CL:WHEN BYTECOUNTVAR
        [SET BYTECOUNTVAR (IDIFFERENCE (OR BYTECOUNTVAL (\EVALV1 BYTECOUNTVAR *BYTECOUNTER*))
          CODE]))
```

(\PEEKCCODE

```
[LAMBDA (STREAM NOERROR) ; Edited 30-Jun-2022 10:03 by rmk
; Edited 27-Jun-2021 23:26 by rmk:
  (CL:FUNCALL (ffetch (STREAM PEEKCCODEFN) of STREAM)
    STREAM NOERROR])
```

(\PEEKCCODE.EOLC

```
[LAMBDA (STREAM NOERROR EOL) ; Edited 30-Jun-2022 10:03 by rmk
; Edited 14-Jun-2021 12:40 by rmk:
  (\CHECKEOLC (CL:FUNCALL (ffetch (STREAM PEEKCCODEFN) of STREAM)
    STREAM NOERROR)
    EOL STREAM T])
```

(\INCCODE.EOLC

```
[LAMBDA (STREAM EOLC BYTECOUNTVAR BYTECOUNTVAL) ; Edited 24-Jul-2022 13:15 by rmk
; Edited 30-Jun-2022 10:12 by rmk
; Edited 8-Aug-2021 14:52 by rmk:
```

:: EOL conversion around essentially a copy of \INCCODE but avoids the extra function call.

:: EOLC of ANY means all patterns go to EOL

:: For simplicity of the raft of implementation functions, they only need to set the constant *BYTECOUNTER* to the number of bytes moved forward
:: (if COUNTP is T). We take responsibility for translating the forward bytes to the countdown value of the particular BYTECOUNTVAR. The goal
:: is simplicity on both sides of the interface.

```
(CL:UNLESS EOLC
  (SETQ EOLC (FFETCH (STREAM EOLCONVENTION) OF STREAM)))
```

;; The implementation functions could do a multiple value return, but that seems to be slower and buggy in current implementation, compared to
;; free-variable setting.

```
(IF BYTECOUNTVAR
  THEN (LET (*BYTECOUNTER* CODE)
        (DECLARE (SPECVARS *BYTECOUNTER*))
        (SETQ CODE (\CHECKEOLC (CL:FUNCALL (ffetch (STREAM INCCODEFN) of STREAM
                                              STREAM T)
                                          EOLC STREAM NIL T)))
        (SET BYTECOUNTVAR (IDIFFERENCE (OR BYTECOUNTVAL (\EVALV1 BYTECOUNTVAR)
                                             *BYTECOUNTER*))
                          CODE)
        ELSE (\CHECKEOLC (CL:FUNCALL (ffetch (STREAM INCCODEFN) of STREAM
                                             STREAM)
                              EOLC STREAM]))
```

(\FORMATBYTESTREAM

[LAMBDA (STREAM BYTESTREAM)

;; Edited 19-Mar-2024 15:57 by rmk: Remove installation of STREAM's EXTERNALFORMAT into BYTESTREAM. That should be done by the
;; formats FORMATBYTESTREAMFN. In particular,
;; Edited 19-Mar-2024 14:31 by rmk
;; Edited 24-Jul-2022 08:30 by rmk: STREAM can be the external format to be used for BYTESTREAM, not just a carrier of that format
;; Edited 22-Jun-2022 11:09 by rmk
;; Edited 24-Jun-2021 17:26 by rmk:
;; Create or modify a stream that will simulate the current character input/output byte sequences of STREAM. The set up here does what is
;; common to all formats: an IO stream starting with STREAM external format and EOL.
;; If the format has its own FORMATBYTESTREAMFN function, that is applied to copy any other state. (Currently that function is a property of the
;; format, not carried over into a stream field that can be changed dynamically.)

```
(CL:UNLESS (AND (STREAMP BYTESTREAM)
                (\IODEP BYTESTREAM 'BOTH))
  (SETQ BYTESTREAM (OPENSTREAM '{NODIRCORE} 'BOTH)))
(LET (FORMAT EOLC)
  (IF (TYPE? STREAM STREAM)
    THEN (SETQ FORMAT (FFETCH (STREAM EXTERNALFORMAT) OF STREAM))
          (SETQ EOLC (FFETCH (STREAM EOLCONVENTION) OF STREAM))
          (CL:WHEN (EQ EOLC ANY.EOLC)
                    (SETQ EOLC (OR (FFETCH (EXTERNALFORMAT EOL) OF FORMAT)
                                   LF.EOLC)))
    ELSEIF (TYPE? EXTERNALFORMAT STREAM)
    THEN (SETQ FORMAT STREAM)
          (SETQ EOLC (FFETCH (EXTERNALFORMAT EOL) OF FORMAT)))
  (REPLACE (STREAM EOLCONVENTION) OF BYTESTREAM WITH EOLC)
  (\SETFILEPTR BYTESTREAM 0)
  ;; Presumably any format-specific cleanup function will know what to do if it receives a format instead of a stream.
  (CL:WHEN (FFETCH (EXTERNALFORMAT FORMATBYTESTREAMFN) OF FORMAT)
    (SETQ BYTESTREAM (APPLY* (FFETCH (EXTERNALFORMAT FORMATBYTESTREAMFN) OF FORMAT)
                              STREAM BYTESTREAM)))
  ;; Maybe the function said no?
  (CL:WHEN BYTESTREAM
    (freplace (STREAM ENDOFSTREAMOP) of BYTESTREAM with (FUNCTION NIL)))
  BYTESTREAM])
```

(\FORMATBYTESTRING

[LAMBDA (STREAM STRING)

; Edited 19-Mar-2024 18:24 by rmk
; Edited 10-Jul-2022 16:39 by rmk
; Edited 22-Jun-2022 11:07 by rmk
; Edited 18-Jun-2022 22:04 by rmk

```
(WITH-RESOURCE \FORMATBYTESTRING.STREAM (\SETFILEPTR \FORMATBYTESTRING.STREAM 0)
  (LET [FSTRING NBYTES (BYTESTRINGFN (FETCH (EXTERNALFORMAT FORMATBYTESTRINGFN)
                                             OF (FETCH (STREAM EXTERNALFORMAT) OF STREAM)
                                         (IF BYTESTRINGFN
                                           THEN (CL:WHEN (SETQ FSTRING (APPLY* BYTESTRINGFN STREAM STRING \FORMATBYTESTRING.STREAM))
                                                         (MKSTRING FSTRING))
                                           ELSE (\FORMATBYTESTREAM STREAM \FORMATBYTESTRING.STREAM)
                                         (FOR C INPNAME STRING DO (\OUTCHAR \FORMATBYTESTRING.STREAM C))
                                         (SETQ NBYTES (\GETFILEPTR \FORMATBYTESTRING.STREAM))
                                         (\SETFILEPTR \FORMATBYTESTRING.STREAM 0)
                                         (SETQ FSTRING (ALLOCSTRING NBYTES))
                                         (FOR I FROM 1 TO NBYTES DO (RPLCHARCODE FSTRING I (\BIN \FORMATBYTESTRING.STREAM)))
                                         FSTRING]))
```

(\CHECKEOLC.CRLF

[LAMBDA (STREAM PEEKBINFLG COUNTP EOLC)

; Edited 6-Dec-2023 23:39 by rmk
; Edited 17-Oct-2023 11:56 by rmk
; Edited 6-Aug-2021 23:30 by rmk:

;; This is called only when a CR has been read and EOLC is either ANY or CRLF. This returns EOL if the next code is an LF
;; If COUNTP, that sets *BYTECOUNTER* freely with the number of LF bytes.

```

(DECLARE (USEDFREE *BYTECOUNTER*))
(LET (CH)
  [SETQ CH (COND
    [PEEKBINFLG
      ;; T from PEEKC. In this case, must leave the fileptr where it was.
      ;; The CR itself hasn't been read, just peeked. So here we have to read it, then peek at the next character to see if
      ;; it is an LF, and then back out the CR
      (COND
        ([EQ (CHARCODE LF)
          (UNINTERRUPTABLY
            ;; Since we are going to \BACKCCODE back the peeked character, we don't need to update the
            ;; counter variable
            (\INCCODE STREAM)
            (PROG1 (\PEEKCCODE STREAM T 'NOEOLC)
              ;; This has to be a call to \PEEKCCODE that doesn't itself to the checkeolc
              ;; LF must be the next char after the CR. We back up over the CR that \INCCODE just read.
              (\BACKCCODE STREAM)))])
          ;; Got the CRLF, it's an EOL
          (CHARCODE EOL))
        (T (SELECTC EOLC
          ((LIST ANY.EOLC 'ANY)
            (CHARCODE EOL))
          (CHARCODE CR]
      ((EQ (CHARCODE LF)
        (\PEEKCCODE STREAM T 'NOEOLC))
      ;; Since we aren't peeking, the CR has actually been read, and we are entitled to read the LF that we just peeked at.
      (IF COUNTP
        THEN (LET (NUMLFBYTES)
          (DECLARE (SPECVARS NUMLFBYTES))
          (\INCCODE STREAM 'NUMLFBYTES 0)
          (ADD *BYTECOUNTER* NUMLFBYTES))
        ELSE (\INCCODE STREAM))
      (CHARCODE EOL))
      (T (SELECTC EOLC
        ((LIST ANY.EOLC 'ANY)
          (CHARCODE EOL))
        (CHARCODE CR]
      CH])
    )
  )
(DECLARE%: DOEVAL@COMPILE DONTCOPY

;; FOLLOWING DEFINITIONS EXPORTED

(DECLARE%: EVAL@COMPILE

(PUTPROPS \CHECKEOLC MACRO (OPENLAMBDA (CH EOLC STRM PEEKBINFLG COUNTP)
  (SELCHARQ CH
    (LF (SELECTC (OR EOLC (FFETCH (STREAM EOLCONVENTION) OF STRM))
      ((LIST LF.EOLC ANY.EOLC 'LF 'ANY)
        (CHARCODE EOL))
        (CHARCODE LF)))
    (CR (SELECTC (OR EOLC (FFETCH (STREAM EOLCONVENTION) OF STRM))
      ((LIST CR.EOLC 'CR)
        (CHARCODE EOL))
        ((LIST ANY.EOLC CRLF.EOLC 'CRLF 'ANY)
          (\CHECKEOLC.CRLF STRM PEEKBINFLG COUNTP EOLC))
        (CHARCODE CR)))
    CH)))
  )
)

;; END EXPORTED DEFINITIONS

(DECLARE%: EVAL@COMPILE

[PUTDEF '\FORMATBYTESTRING.STREAM 'RESOURCES ' (NEW (OPENSTREAM '{NODIRCORE} 'BOTH)
)
)

(/SETTOPVAL '\FORMATBYTESTRING.STREAM.GLOBALRESOURCE NIL)

;; NULL device, from FILEIO

(DEFINEQ

(\NULLDEVICE
  [LAMBDA NIL
    ;; Defines the NULL device, an infinite source or sink
    (* bvm%: "30-Jan-85 22:06")
  ]
)

```

```
(\DEFINEDEVICE 'NULL (create FDEV
    DEVICENAME _ 'NULL
    RANDOMACCESSP _ T
    NODIRECTORIES _ T
    CLOSEFILE _ (FUNCTION NIL)
    DELETEFILE _ (FUNCTION NIL)
    OPENFILE _ (FUNCTION \NULL.OPENFILE)
    REOPENFILE _ (FUNCTION \NULL.OPENFILE)
    BIN _ (FUNCTION \EOF.ACTION)
    BOUT _ (FUNCTION NIL)
    PEEKBIN _ [FUNCTION (LAMBDA (STREAM NOERRORFLG)
        (AND (NULL NOERRORFLG)
            (BIN STREAM))

    READP _ (FUNCTION NIL)
    BACKFILEPTR _ (FUNCTION NIL)
    EOFP _ (FUNCTION TRUE)
    RENAMEFILE _ (FUNCTION NIL)
    GETFILENAME _ (FUNCTION NIL)
    EVENTFN _ (FUNCTION NIL)
    BLOCKIN _ (FUNCTION \EOF.ACTION)
    BLOCKOUT _ (FUNCTION NIL)
    GENERATEFILES _ (FUNCTION \NULLFILEGENERATOR)
    GETFILEPTR _ (FUNCTION ZERO)
    GETEOFPTR _ (FUNCTION ZERO)
    SETFILEPTR _ (FUNCTION NIL)
    GETFILEINFO _ (FUNCTION NIL)
    SETFILEINFO _ (FUNCTION NIL)
    SETEOFPTR _ (FUNCTION NIL))
```

(\NULL.OPENFILE

```
[LAMBDA (NAME ACCESS RECOG PARAMETERS DEVICE OLDSTREAM) (* bvm%: "30-Jan-85 22:05")
  (OR OLDSTREAM (create STREAM
    USERCLOSEABLE _ T
    ACCESS _ ACCESS
    FULLFILENAME _ NIL
    DEVICE _ DEVICE])
)
```

(DECLARE%: DONTEVAL@LOAD DOCOPY

(\NULLDEVICE)

:: Also from FILEIO, but not clear that this is or ever has been used.

(DEFINEQ

(\CREATE.THROUGH.EXTERNALFORMAT

```
[LAMBDA NIL ; Edited 24-Jul-2022 08:08 by rmk
; Edited 23-Jun-2021 13:34 by rmk:
```

:: Create the :THROUGH external format. EOL is adjusted so that the .EOLC callers will not do any conversion.

```
(MAKE-EXTERNALFORMAT :THROUGH (FUNCTION \THROUGHIN)
  (FUNCTION \PEEKBIN)
  (FUNCTION \THROUGHBACKCCODE)
  (FUNCTION \THROUGHOUTCHARFN)
  NIL
  (CL:IF (EQ (CHARCODE CR)
    (CHARCODE EOL))
    CR.EOLC
    LF.EOLC)
  NIL
  (FUNCTION (LAMBDA (STREAM STRING)
    (MKSTRING STRING))
```

(\THROUGHIN

```
[LAMBDA (STREAM COUNTP) ; Edited 6-Aug-2021 16:13 by rmk:
```

:: Read in a single byte from STREAM and returns it without any character conversion, just through as if.

:: If COUNTP is non-NIL, the byte counter is always set to 1.

```
(DECLARE (USEDFREE *BYTECOUNTER*))
(CL:WHEN COUNTP (SETQ *BYTECOUNTER* 1))
(\BIN STREAM])
```

(\THROUGHBACKCCODE

```
[LAMBDA (STREAM COUNTP) ; Edited 6-Aug-2021 16:14 by rmk:
  (DECLARE (USEDFREE *BYTECOUNTER*))
  (CL:WHEN (\BACKFILEPTR STREAM)
    (CL:WHEN COUNTP (SETQ *BYTECOUNTER* -1))
  T)])
```


(\THROUGHOUTCHARFN
[LAMBDA (OUTSTREAM CHARCODE)

; Edited 26-Feb-91 13:44 by nm

;;; Encoder for THROUGH format.

```
(COND
  ((> CHARCODE 255)
   (\BOUT OUTSTREAM (\CHARSET CHARCODE))
   (\BOUT OUTSTREAM (\CHAR8CODE CHARCODE)))
  (T (\BOUT OUTSTREAM CHARCODE]))
)

(DECLARE%: DONTEVAL@LOAD DOCOPY

(\CREATE.THROUGH.EXTERNALFORMAT)
)
```

FUNCTION INDEX

FIND-FORMAT	4	\EXTERNALFORMAT.DEFPRINT	3	\OUTCHAR	4
MAKE-EXTERNALFORMAT	2	\FORMATBYTESTREAM	6	\PEEKCCODE	5
SYSTEM-EXTERNALFORMAT	4	\FORMATBYTESTRING	6	\PEEKCCODE.EOLC	5
\BACKCCODE	4	\INCCODE	4	\REMOVE.EXTERNALFORMAT	3
\BACKCCODE.EOLC	5	\INCCODE.EOLC	5	\THROUGHBACKCCODE	8
\CHECKEOLC.CRLF	6	\INSTALL.EXTERNALFORMAT	3	\THROUGHIN	8
\CREATE.THROUGH.EXTERNALFORMAT ..	8	\NULL.OPENFILE	8	\THROUGHOUTCHARFN	9
\EXTERNALFORMAT	2	\NULLDEVICE	7		

VARIABLE INDEX

DEFAULT-EXTERNALFORMAT	4	*EXTERNALFORMATS*	4	SYSTEMRECLST	2
--------------------------------	---	-------------------------	---	--------------------	---

MACRO INDEX

\CHECKEOLC	7
------------------	---

RECORD INDEX

EXTERNALFORMAT	1
----------------------	---
