KOTO RELEASE NOTES – ADDENDA

This document includes some additional information and clarification inadvertently omitted from the Koto Release Notes. It also lists some problems that have been identified by early testers of the Koto release, and, where possible, workarounds to avoid the problems. Please insert these pages into your new Release Notes manual.

Features

- Floppy Support for 1186 Workstations
- Fonts
- Keyboard
- File Package

Floppy Support for 1186 Workstations

This section describes Interlisp-D's file device {Floppy}, which is used to interact with floppy drives on Xerox 1186 machines.

Use a $5\frac{1}{4}$ inch double-sided, double-density, soft sectored floppy disk. Insert the floppy into the floppy drive face up, the edge of the floppy with two $1\frac{1}{2}$ inch-long cover holes (one hole per side) going in first. The write-protect notch will be on the left edge as you look down on the floppy from above. You write protect a floppy by covering the write-protect notch with the tabs provided with the floppies. (This is exactly opposite of the 1108.)

In general, Interlisp-D can be used to read, write, and otherwise interact with files on Pilot-formatted floppy disks through standard Interlisp input/output functions. All familiar operations such as LOAD, OPENFILE, READ, PRINT, BIN, BOUT, GETFILEINFO, SETFILEINFO, GETFILEPTR, SETFILEPTR, etc., work with floppies. COPYFILE is commonly used to archive and unarchive files between floppies and other file devices. Files on floppies can be compiled, edited, hard-copied, etc., just as files may be on all other ordinary file devices.

Naming, Erasing, and Formatting Floppies

To erase and establish track information on a floppy, especially a floppy that is brand-new, call
(FLOPPY.FORMAT *NAME AUTOCONFIRMFLG SLOWFLG*)
[Function]

The *NAME* argument becomes the name of your floppy. It can be any string or atom of 106 or fewer characters. It's a good idea to label the outside of your floppy with the same name using a sticky label and a soft marker.

AUTOCONFIRMFLG controls questioning you about destroying the contents of a floppy that appears to contain valid information. If *AUTOCONFIRMFLG* is T, you will not be asked to confirm.

SLOWFLG determines how thorough a formatting job is to be done on the floppy you format. If *SLOWFLG* is T, FLOPPY.FORMAT completely erases your floppy, setting down track information and critical Pilot records on it. When *SLOWFLG* is left NIL, only the Pilot records needed to give your floppy an empty directory are written. Use *SLOWFLG* = T with a brand-new floppy.

Typically you will leave *AUTOCONFIRMFLG* and *SLOWFLG* as NIL. For example, doing (FLOPPY.FORMAT "My Floppy") will create a blank floppy named "My Floppy."

To reset the name put onto a floppy by FLOPPY.FORMAT, use

(FLOPPY.NAME *NAME*) [Function]

If *NAME* is NIL, then FLOPPY.NAME reads the name put onto a floppy by FLOPPY.FORMAT or FLOPPY.NAME.

Copying Floppies

The ability to copy Pilot floppies is supported through two functions:

(FLOPPY.TO.FILE *TOFILE*) [Function]

(FLOPPY.FROM.FILE *FROMFILE*) [Function]

FLOPPY.TO.FILE copies the contents of the current floppy to a file, and FLOPPY.FROM.FILE is FLOPPY.TO.FILE's inverse. For instance, to copy the contents of one floppy onto another, insert the floppy to be copied into the floppy drive. Then type (FLOPPY.TO.FILE 'ToFile). Remove the first floppy and insert a blank floppy. Type (FLOPPY.FROM.FILE 'FromFile). The first floppy has now been copied.

The *TOFILE* produced by FLOPPY.TO.FILE is approximately 700 pages long and can be placed on a file server or a logical volume of your machine. FLOPPY.FROM.FILE can be used more than once if you would like to make more than one copy. As an alternative to using FLOPPY.SCAVENGE (described below), the *TOFILE* produced by FLOPPY.TO.FILE can be usefully edited to salvage the contents of a floppy that has been damaged.

Loading Sysouts and Other Large Files Onto Floppies

Sysouts may be created on floppies through Interlisp's SYSOUT or MAKESYS functions, then later installed on the same or another Xerox 1186 using the Installation Utility. Sysouts may also be taken from other file devices and put onto floppies through the use of the function FLOPPY.MODE (see below). To copy a sysout to floppies, simply do one of the following functions:

(SYSOUT '{FLOPPY}) [Function]

(MAKESYS '{FLOPPY}) [Function]

You are prompted to insert new floppies as they are needed. It generally takes at least ten floppies to store a sysout. To load in a sysout from floppies, you can use the System Tool.

The normal mode of operation for {FLOPPY} is

(FLOPPY.MODE 'PILOT) [Function]

Two special modes of operation for floppies, SYSOUT, and HUGEPILOT are also available. You can put floppies into either of the two modes to copy sysouts or huge files from file servers to floppies or from floppies to file servers.

In SYSOUT mode, you may use COPYFILE to move a sysout off another file device onto floppies. To do this, you must first set the floppies into SYSOUT mode, then do the COPYFILE. For example,

```
(FLOPPY.MODE 'SYSOUT)

(COPYFILE '{FileServer}<Directory>LISP.SYSOUT '{FLOPPY})
```

will put the current Interlisp-D LISP.SYSOUT onto floppies. While in SYSOUT mode, you can copy as many sysouts as you like onto floppies. The *Installation Utility* is then used to load these sysouts onto a Xerox 1186. Similarly, sysouts can be copied from floppies onto another file device using COPYFILE when in SYSOUT mode. For example,

```
(FLOPPY.MODE 'SYSOUT)
(COPYFILE '{FLOPPY} '{YourFileServer}<YourDirectory>
                                        Your.Sysout)
```

will copy the sysout onto your directory on a file server. To get back to ordinary floppy operation, type (FLOPPY.MODE 'PILOT).

You can write and read huge Pilot files onto multiple floppies in HUGEPILOT mode, which can be set by typing (FLOPPY.MODE 'HUGEPILOT). This mode is practically identical to SYSOUT mode, with the exception that you have control over the names of files and floppies.

Manipulating Floppy Space

The following function tells you how much space is left on a floppy.

(FLOPPY.FREE.PAGES) [Function]

returns the number of free pages on the current floppy. Pilot floppy files are contiguously represented on a floppy disk. If you are using your floppy interactively (not just doing a simple series of COPYFILE commands after a FLOPPY.FORMAT), don't cram your floppy to capacity. Try to keep such a floppy less than 75 percent full.

Testing Whether a Floppy is in the Drive

Two functions are available for testing whether a floppy is in the floppy drive.

(FLOPPY.CAN.READP) [Function]

tests if there is a floppy in the floppy drive. FLOPPY.CAN.READP does not provide any debouncing (protection against not fully closing the floppy drive door).

(FLOPPY.CAN.WRITEP) [Function]

tests if there is a floppy in the floppy drive and the floppy drive can write on this floppy. (The floppy drive can only write on floppies whose write-protect notches are not covered with tape.)

Note: Unlike the 1108, the 1186 can not tell if the floppy drive door has been opened.

Scavenging Floppies

(FLOPPY.SCAVENGE) [Function]

attempts to repair a floppy whose critical records have become confused. Also, if you accidentally delete floppy files you shouldn't have deleted, FLOPPY.SCAVENGE retrieves them (provided you don't wait till after they have been overwritten by new files).

Fonts

Instructions for installing Fonts from the Display Font and Font Metrics Floppy Disks

The Display Fonts floppy disks contain files of font information for use with the workstation display; the Font Metrics floppies contain character-width information that is used in formatting text for printing.

You will need to copy some of the font files from each set of floppies onto either your workstation's disk or a file server. Then, you'll need to inform Lisp of where you put the files by setting the variables DISPLAYFONTDIRECTORIES and INTERPRESSFONTDIRECTORIES, the former for the display fonts, the latter for the printer font metrics. Each of these variables is a list of one or more directory specifications, and is most conveniently set in your personal or site init file. For example, if you copied the fonts to the FONTS subdirectory on your local disk, you might set each of these variables to ({DSK}<LISPFILES>FONTS>). If you stored them on the Fonts directory of the server {YOURSERVER}, you could set the variables to be ({YOURSERVER}<Fonts>). You might choose to store the display fonts and the printer metrics in different places, in which case DISPLAYFONTDIRECTORIES and INTERPRESSFONTDIRECTORIES would, of course, have to have different values.

For normal Interlisp-D operation, you will need at least these files:

Display Fonts	Printer Metrics
Terminal8-C0.DisplayFont	Terminal8-C0.WD
Terminal8-C356.DisplayFont	Terminal8-C356.WD
Terminal8-C357.DisplayFont	Terminal8-C357.WD
Modern10-C0.Displayfont	Modern10-C0.WD
Modern10-C356.DisplayFont	Modern10-C356.WD
Modern10-C357.DisplayFont	Modern10-C357.WD
Modern8-C0.DisplayFont	Modern8-C0.WD
Modern8-C356.DisplayFont	Modern8-C356.WD
Modern8-C357.DisplayFont	Modern8-C357.WD
Modern8-I-C0.DisplayFont	Modern8-I-C0.WD
Modern8-I-C356.DisplayFont	Modern8-I-C356.WD
Modern8-I-C357.DisplayFont	Modern8-I-C357.WD
Gacha10-C0.DisplayFont	Gacha10-C0.WD

However, we recommend that you copy all of the following font files:

Display Fonts	Printer Metrics
*-C0.DisplayFont	*-C0.WD
*-C356.DisplayFont	*-C356.WD
*-C357.DisplayFont	*-C357.WD
*-C360.DisplayFont	*-C360.WD
*-C361.DisplayFont	*-C361.WD

The recommended set of fonts includes all the characters necessary to type most European languages, including accents and accented characters. If you wish to use characters from other languages, you will need to copy the following files:

Language	Display Fonts	Printer Metrics
Greek	*-C46.DisplayFont	*-C46.WD
Russian	*-C47.DisplayFont	*-C47.WD
Hiragana	*-C44.DisplayFont	*-C44.WD
Katakana	*-C45.DisplayFont	*-C45.WD

If you wish to type Japanese or Chinese Kanji, you will need to copy all of the font files.

File Package

The user's WHENCHANGEDFN is now called with REASON being either CHANGED or DEFINED, instead of NIL or T.

Any user code that was testing for the specific values NIL/T must be changed to look for CHANGED/DEFINED. (Ref. page 31 of the Koto Release Notes)

Keyboard

The HELP interrupt character is now ↑ G by default, instead of ↑ H.

↑ H has been re-assigned to its ASCII-standard meaning of "backspace."

Bugs

- Interlisp-D Environment
- File System
- Fonts
- Graphics
- Operating System
- Keyboard

Interlisp-D Environment

Problem: Idle Mode Cursor can be lost if you break during Idle.

When in Idle mode, the cursor is changed to an all-blank cursor, so that all you see on your screen is whatever idling pattern you have chosen. When you exit from Idle, the previous cursor is restored. However, if some time after Idle mode is entered a program runs that saves the current cursor, to be restored on its completion, and that program does not finish until Idle mode is exited, then it will "restore" a blank cursor. The most common case of this is a program that falls into a break window while in Idle mode, since break windows save and restore the cursor.

Workaround: If you find yourself without a cursor, evaluate (CURSOR T) in your executive window. Do not do this in a break window, since your effort will be undone after you exit the break. Alternatively, pass your cursor over a window, such as TEdit, that changes the cursor without ever restoring the old one.

File System

Don't use SEE on binary files.

The function SEE and the File Browser's SEE command assume the file you're viewing is a text file. As a result, it interprets certain sequences of bytes as representing run-coding information for the NS extended character set. At best, the system will spend time searching for the necessary fonts to display the special characters that SEE mistakenly believes are in the file; at worst, you will get a break with error message "24-bit extended NS encoding not supported" if it encounters an escape to 24-bit encoding among the random bytes it reads. In the Intermezzo release this was not a problem, because there was always a one-to-one mapping of byte to character.

Thus, to view binary files that might contain interesting text imbedded in them, use the File Browser's Unformatted SEE command, or use (COPYFILE file T) from the Interlisp-D Executive window.

Floppy file names containing only subdirectories not supported.

In previous releases, it was possible to name files on {floppy} with malformed names of the form "{floppy}subdir1>subdir2>name". In Koto, if you are connected to {floppy} and specify a name "subdir1>subdir2>name", the file system will coerce to the well-formed name "{floppy}<subdir1>subdir2>name", which will not succeed if you had previously written a file there with the malformed name.

COPYFILE and the Library Package File Cacher do not coordinate

If you have the Library Package File Cacher turned on, you can get an assortment of errors where the cacher does not interlock with functions that explicitly avoid the cache. As a particular example, the function COPYFILE does not use the cache (since you usually aren't planning to access the copied file soon, you just want to move it around). As a result, if you call COPYFILE on a cached file immediately after having closed the file after writing it, you may get the wrong version of it, or no version at all (a FILE NOT FOUND error). Worse yet, if you call COPYFILE while the cacher is writing out the file to the server, you may interfere with each other.

Fonts

Occasionally a package will require a font which for some reason is not available. When this happens, a break window appears with the message "font not found", and the required font is then listed. The package may be aborted with the up-arrow key, or a different font may be substituted instead. If the requested font was for the display, the default display font may be substituted by entering to the break,

(RETFROM 'FONTCREATE (DEFAULTFONT 'DISPLAY)).

Since this is equivalent to performing a font substitution other than what the package intended, the resulting formatting may be unpredictable.

Graphics

Set global variable PRINTSERVICE to floating point value 9.0.

When using FILLPOLYGON to Xerox 8044 Interpress printers, the global variable PRINTSERVICE must be set to the same value as the Print Service installed on your printer, currently either 8.0 or 9.0. Thus, if your printer is running Print Service 9.0, you must set the global variable PRINTSERVICE to the floating point value 9.0. This works around an incompatible change in the Xerox 8044 Interpress implementation.

Operating System

1186 Floppy disk bug

Problem: The 1186 floppy disk drive can get into an error state in which the Lisp code accessing the floppy hangs forever with no error message. Sometimes this state is accompanied by a particular noise from the floppy drive itself.

Workaround: Interrupt the process, using either the Control-B or Control-G interrupt, remove the floppy disk, reinsert the floppy, then type OK in the break window.

Library Packages

- File Browser
- HELPSYS
- GRAPHER
- CMLARRAY
- CML
- DECL
- RS232
- NSChat

File Browser

Caution when using File Browser's Delver on Unix.

Some versions of the Vax Unix Pup FTP server enumerate the newest version of a file twice: once with its correct version number and once without any version number. As a result, if you use the File Browser's "Delete Old Versions" command trimmed to one version, the File Browser will attempt to delete one of those two files, which seems to cause both to vanish! Therefore, when using File Browser to Unix and applying the "Delete Old Versions" command, always tell it to retain one more version than you otherwise would have.

File Browser hardcopy bug

If you select the Hardcopy command in the File Browser when no files are selected, the File Browser erroneously calls the function LISTFILES with no arguments. This causes all files in the list NOTLISTEDFILES (the set of files believed by the file package to be updated but not yet listed) to be hardcopied.

HELPSYS

Note for 1186 users:

The file IRM.HASHFILE requires three 5 $\frac{1}{4}$ inch floppies (HelpSys #7, HelpSys #8, and HelpSys#9.) To copy this file from floppies, you must use HUGEPILOT mode. For those unfamiliar with HUGEPILOT mode, type the following into the Interlisp-D Executive window:

```
(FLOPPY.MODE 'HUGEPILOT)

(COPYFILE '{FLOPPY}IRM.HASHFILE '{server}<directory>
                                        IRM.HASHFILE)

(FLOPPY.MODE 'PILOT)
```

When it is time to insert the next floppy (IRM.HASHFILE#2 = HelpSys#8, IRM.HASHFILE#3 = HelpSys#9), you will be prompted in the Interlisp-D Executive window. It is important to note that a random access device must be used for HelpSys to function correctly.

Clarification of IRM functions

In the HELPSYS Library package, add the following clarification to the functions IRM.LOOKUP and IRM.SMART.LOOKUP: The WINDOW argument, if non-NIL, must have a window property MONITORLOCK whose value is a MONITORLOCK; the lock is used to synchronize possible multiple users of the window. The TYPE argument, if specified, must be an atom spelled exactly as the corresponding bracketed type description in the Interlisp Reference Manual, including case, e.g., Function, Variable, I.S.Operator.

GRAPHER

The Grapher package has the following new features:

There is a new capability on the edit menu that allows the user to change the label of a node. The default method for changing the label is to prompt the user for a string or atom for the new label. This behavior can be changed by supplying a function in the GRAPH.CHANGELABELFN field of the GRAPH record.

The graph edit menu can now be redefined by the user. There is a new variable EDITGRAPHMENUCOMMANDS from which the graph edit menu is constructed. To add new commands, change EDITGRAPHMENUCOMMANDS, and set the variable EDITGRAPHMENU to NIL. The next time the editing facilities are invoked, the menu will be rebuilt. The forms defining the actions of new menu items can refer freely to the variable GRAPHWINDOW, which is bound to the window in which the graph resides.

The code for representing a node as a little black box was fixed to take the image stream's DSPSCALE into account. Previously, the little black box was invisible on Interpress and Press pages.

CMLARRAY

The "fast array accessors and settors" available in a previous release of CMLARRAY do not work. Please use AREF and (SETF (AREF...) ...) instead. Thus, if your code uses these "fast functions," you will have to change them to the generic calls.

Bit vector printing is incorrect. EQ tail elements are removed, but the total length is not printed.

CMLArray does not automatically load the CMLArrayInspector package. CML will load both the array package and its inspector. If you're using the CMLArray package without CML you must explicitly load CMLArrayInspector.

CML

As mentioned in the CML Library Package documentation, only a small part of Common Lisp has been implemented. Beware of Interlisp functions with similar names and different semantics. The coming package system will alleviate this problem.

Characters

The character predicate CHARACTERP does not work.

NAME-CHAR signals an error if given an invalid character name.

CMLExec

This package currently does not use the CMLRDTBL; atoms must be entered in lowercase to be correctly translated into uppercase CL: prefix symbols.

Type specifiers

Some cases of TYPEP do not work, including:

ARRAY

BIGNUM

CHARACTER

STRING-CHARACTER

General

INCF always fails by signaling an error.

DOLIST ignores its *RESULTFORM*.

CATCH's compiler macro will signal an error at compile time if the *TAG* is a constant. To eliminate this do (REMPROP 'CATCH 'DMACRO).

TYPECASE does not handle the OTHERWISE clause. Use T instead.

DEFMACRO's &AUX lambda list option is not implemented.

The DMACRO for CL:APPLY does not work when the form is in a context in which the value of the CL:APPLY is not needed.

Reader

The # feature reader macros # + and #- do not work.

Structures

DEFAULT-STRUCTURE-PRINTER will always print to the current stream (NIL), ignoring its *FILE* argument.

DECL

Compiling DECL forms causes ARG NOT LIST.

There is a bad interaction between the DECL Library package and the compiler that causes an error ARG NOT LIST when compiling DECL forms. Setting CAR/CDRERR to NIL will inhibit this error.

RS232

The current release of the 1186's RS232 package is entirely compatible with the 1108's RS232 package, as documented in the Lisp Library Packages documentation set. Both the RS232 port (labeled "DTE") and the TTY port (labeled "DCE") are supported. As with the 1108, support for the RS232 port is embodied in DLRS232C.DCOM, and support for the TTY port is include in DLTTY.DCOM.

However, the following problems are known to exist:

1. The "frame timeout" feature is not currently implemented by the RS232 handler in the I/O processor. As a result, RS232 performance in interactive applications, such as Chat, will probably be poor; bursts of characters will not be delivered to Lisp (e.g., appear in the Chat window) until about one second after the end of each burst. This problem is expected to be solved soon, and fixes will be delivered to customers in the form of a new Lisp microcode file.

2. It is not currently possible to connect the 1186's RS232 port to the RS232 port on either another 1186 or an 1108. However, it does appear to be possible to connect an 1186 to another 1186 or 1108 using the 1186's TTY port.

   No problems have been encountered in connections between 1186s and modems or printers. The inability to connect 1186s together is probably due to incorrect handling of low-level RS232 signals, and is being investigated.

3. The maximum suggested baud rate on the 1186 machine is 9600 baud.

There is a difference between the 1108 and 1185/1186 processors in the mutual setting of the "parity" and "character length" parameters for the RS232 and TTY ports. The following table illustrates the most likely settings for these parameters:

Parity Option	1108	1185/1186
No parity	Parity NONE, 7- or 8- bit chars	Parity NONE, 7- or 8- bit chars
Even parity	Even, 7- bit chars	Even, 8- bit chars
Odd parity	Odd, 7- bit chars	Odd, 8- bit chars

NSChat

Problem: In the automatic login of NSChat to the Xerox Interactive Terminal Service, a "Q" appears before the Logon command, resulting in an invalid "QLogon," with users' passwords being echoed in plain text in the Chat window.

Workaround: A patch to the NETWORKLOGINFO variable will prevent this from happening.

After Chat.DCOM has been loaded, modify the NS LOGIN entry to look as follows:

(NS (LOGIN ↑H "LOGON" CR USERNAME CR USERNAME PASSWORD CR))

where ↑H is a newly-added, single character atom generated by evaluating

(CHARACTER 8).

KOTO RELEASE NOTES–ADDENDA II

This document supplements the Koto Release Notes and the Koto Release Notes Addenda. It contains corrections to previously-issued software and documentation. Please insert these pages into your Release Notes manual. Included in this document is a replacement for page 12 of the Koto Release Notes. This change is explained below under "Features" in the section titled **Keyboard for the 1186 Workstation.** The basic format of the original release notes document is used here. Whenever corrections to release documentation are described, references to appropriate page numbers are provided.

- Features
- Bug Fixes
- Library Packages

Features

The following features are addressed in this section:

- Fonts
- Middle Mouse Button for the1186 Workstation
- Keyboard for the 1186 Workstation
- Interlisp-D Environment
- Floppy Support for the 1186 Workstation
- 1108 Workstation System Tools

Fonts

Addition to Koto Release Notes – Addenda, pages 6-7

The list of fonts to copy should include *-C41.WD files. Documents with hypens will be formatted incorrectly for interpress hardcopy if this font file is not available.

Middle Mouse Button for 1186 Workstation

The new sysout contains a function to enable or disable the action of the middle mouse button.

When the middle mouse button is disabled, "phantom" down transitions of the middle mouse button will no longer occur on 1186 processors equipped with 2-button mice.

(MIDDLEMOUSE FLG) [Function]

MIDDLEMOUSE enables or disables the action of the middle mouse button. If MIDDLEMOUSE is called with no arguments, it returns T if the middle button is enabled, or NIL if it is disabled.

If MIDDLEMOUSE is called with the FLG argument set to T, it enables the middle button. If it is called with FLG set to NIL, it disables the middle button. In either case, it returns the old value of the flag.

Therefore, (MIDDLEMOUSE NIL) will disable the middle mouse button. This is recommended if you are experiencing down transitions of the middle mouse button on an 1186 workstation with a two-button mouse.

Keyboard for the 1186 Workstation

Keyboard assignments for the EDIT, CONTROL and SHIFT keys have been changed.

The keyboard diagram included with this document shows the new positions. This change should be considered temporary. In a future release, these key positions will change to make the keyboard more compatible with both programming and editing applications. Use the new diagram to replace page 12 of your Koto Release Notes.

Interlisp-D Environment

Caution: **Avoid concurrent idle processes.**

It is possible to get more than one Idle process running at the same time. To avoid this behavior, type the following line into the Interlisp-D Executive window:

```
ADVISE(IDLE (AND \IDLING (RETURN]
```

Floppy Support for the 1186 Workstation

CPM mode is not currently implemented for the 1186.

Refer to the 1186 User's Guide for information on floppy support.

1108 Workstation System Tools

The Diagnostics Tool incorrectly formats floppies.

Be sure to use the **Format!** command in the Floppy subwindow when you wish to format floppy disks. You can invoke this subwindow by selecting the **Floppy!** command in the Lisp System Tool window. Note that the **Floppy!** command will ask you to confirm twice before continuing the formatting operation.

Bug Fixes

This section contains a correction to one of the bug fixes documented in the Koto release Notes:

- Window System
- 1108 Local Disk

Window System

Correction to Koto Release Notes document, page 43

The original Release Notes documentation states that when you are using a brush with an even width, DRAWCURVE will place the extra bit "up and/or to the left". The actual behavior, while now consistent between DRAWLINE and DRAWCURVE, is to place the extra bits up and/or to the **right.**

1108 Local Disk

Access provided to local file system on 29 Mb disk drives

The local file system on SA4000 (29Mb) drives can now be accessed. However, label verify on write is turned off on these drives so the file system will not detect data being written to the wrong location on the disk.

DISKPARTITION function works correctly on 1186 workstations

The function DISKPARTITION now returns the correct partition when called on an 1186.

FILENAMEFROMID function works correctly on 1108 and 1186 workstations

The function FILENAMEFROMID, described in the 1108 & 1186 User's Guides, has been fixed. This function is critical for recovering from local disk errors.

Caution: **Allow enough space on your local disk.**

Hardcopying a large document when your local disk is almost full can cause your machine to fall into MP code 9318. This can be avoided if you make sure there is enough space on your local disk for the interpress scratch file + 20 pages. The interpress scratch file is approximately twice the size of the document.

Library Packages

The following supplements Koto Release Notes Library Packages documentation:

- CMLFLOATARRAY
- TCP/IP
- RS232
- DInfo/HelpSys
- RS232CHAT

CMLFLOATARRAY

Complex format arrays must be quadword-aligned.

To create a quadword-aligned array use the :ALIGNMENT 64 keyword argument to MAKE-ARRAY.

Arrays given as arguments to the FFT functions must be page-aligned.

To create a page-aligned array use the :PAGE-ALIGN T keyword argument to MAKE-ARRAY

RS232

RS232 input connections have bounds on buffer space.

Previously there was no limit to the number of packets that could accumulate on the input side of the RS232 device when an input stream was opened to it. As a result, it was possible to consume arbitrary amounts of storage with an RS232 input stream that was not being actively read. This is now fixed.

Opening an RS232 stream for output no longer opens an unnecessary input stream.

(OPENSTREAM '{RS232} 'OUTPUT --) used to open another stream for input as well This was particularly inefficient in light of the previously unbounded input buffering. Only the requested stream is opened now.

RS232SENDBREAK and RS232MODEMHANGUP work on 1186

(FORCEOUTPUT RS232stream T) works on 1186.

Previously, if the WAITFORFINISH argument were true in a call to FORCEOUTPUT on an RS232 stream, FORCEOUTPUT would never return. This has now been fixed.

Calling RS232C.INIT more than once per session no longer crashes 1186s.

Previously, if you called RS232C.INIT more than once between LOGOUTs, you would soon get an error under the garbage collector, most commonly the MP code 9329.

Reinitialization of RS232C port using DLRS232C no longer crashes.

The problem was a 9305 error when turning RS232 on after LOGOUT, if RS232 had been turned off before the LOGOUT

RS232CHAT

The following ADVICE should be installed after RS232CHAT.DCOM is loaded:

```
(ADVISE 'RS232CHAT.SET.PARAMETERS 'BEFORE
    '(ALLOW.BUTTON.EVENTS))
```

This prevents the mouse from being disabled while the "Set Line Parameters" option command is creating the RS232 port settings menu.

DInfo/HelpSys

DInfo and HelpSys no longer lose the device name associated with IRM.HOST&DIR.

TCP/IP

TCP Chat problems with BSD Unix resolved.

There was a flaw in the handling of TCP retransmissions for which the retransmitted packets are not exact duplicates of previously transmitted packets. This is an efficiency enhancement that only Unix and Ultrix implement. Such retransmitted packets would be incorrectly merged into the incoming data stream, causing a 9318 MP code when the data were later consumed. This is now fixed

---

INFILEP of non-existent file on Tops-20 system using TCPFTP now fixed.

The server returns the inappropriate response "? not found", which the Lisp code had not been correctly handling. This response is now correctly handled by the Lisp code.

ICMP redirection messages now being handled properly.

Receipt of an ICMP redirection message sometimes crashed the machine. This has been fixed.

Fragmented IP packets are now correctly reassembled.