Front Cover, to be replaced by Mac version

KERMIT NEWS

Volume 3, Number 1

June, 1988

DRAFT of 11 March 11 14:50 from V3N1.MSS, line 49

Kermit News *is published periodically free of charge by Columbia University, Center for Computing Activities, 612 West 115th Street, New York, NY 10025, USA, phone 212-280-3703.*

Editor: *Christine Gianone*

EDITOR'S NOTES

Christine Gianone
Columbia University, New York City

Welcome to the third issue of Kermit News. This newsletter is published whenever enough news is gathered to report, and time permits. The two previous issues were Volume 1, Number 1 (July 1986) and Volume 2, Number 1 (November 1987). We hope to publish the newsletter more frequently in the future.

The Kermit newsletter attempts to inform Kermit users of new developments and report on some of the varied Kermit uses. Readers who are using Kermit in especially unusual, interesting, or beneficial ways are encouraged to submit articles of about 500 words for publication in future issues. Testimonials from satisfied Kermit users are always welcome too.

Everyone is permitted to copy and redistribute Kermit programs so long as it is not done for profit. If you make fixes, changes, improvements, or write new documentation of general interest, or especially if you develop a Kermit program for a system that didn't have one before, you are encouraged to send your work back to Columbia University so that we can maintain a definitive and comprehensive set of Kermit implementations for further distribution.

Product News

New versions and releases of Kermit programs are continually appearing, as you will see in the sections "Summary of Recent Releases" (page 9) and "Major New Releases" (page 3). In some cases, a single portable version is able to replace several more specific versions, as reported in the "Portable 370 Kermit" article (page 3). Kermit is currently available for over 300 different computers and operating systems.

DEC's large systems, the DECsystem-10 and DECSYSTEM-20, have begun to sail smoothly into eternity. Our own DEC-20, CU20B, home of Kermit Distribution since its beginning in 1981, will be retired this Fall. After that time we will no longer be able to provide DEC-20 DUMPER or DEC-10 BACKUP/Interchange format tapes! If you need tapes in this format, *this is your last chance to order*. This offer expires September 1, 1988.

By popular demand, we have added some new items and formats to our catalog, which you may select on the enclosed order form:

- DEC MicroVAX TK50 tape cartridges. These should also be readable on TK70 drives. Available for both VAX/VMS and Ultrix-32. Cartridge #1 holds Tapes A, B, and E; Cartridge #2 has Tapes C, D, and E.
- DEC MicroVAX RX50 diskettes for VMS and Ultrix. These should also be readable on RX33 drives. Executables, utilities, and documentation for VMS or Ultrix Kermit are included.
- IBM PC MS-DOS Kermit source diskettes available on IBM PC 5 1/4-inch 360K and PS/2 formatted 3.5-inch 720K diskettes.
- UNIX C-Kermit source on IBM PC 5 1/4-inch diskettes. These are especially useful for Xenix users on PC/AT's or similar systems.
- MacKermit C source on Macintosh formatted 3.5-inch diskettes.

Thanks to Digital Equipment Corporation for an equipment grant that allows us to make the MicroVAX TK50s and RX50s, and to International Business Machines for providing a PS/2 for 3.5-inch and 5 1/4-inch diskette production.

Disk Volunteers Needed

At Columbia University, we try to meet the demand for Kermit programs in a variety of tape and diskette formats. Since we don't have one of every kind of computer to make disks and tapes on, we take every opportunity to ask public-spirited volunteers to come forward and make Kermit available on native media for the kinds of computers we don't have: Acorn, Apple II, Apricot, Alpha Micro, Altos, Amstrad, Apollo, AT&T 3Bx, Atari . . . And those are just the A's!

In the Summary of Recent Releases on the inside back cover you will see some of our current diskette and tape volunteers. If you would like to add your name to the list, please contact Kermit Distribution. Even better, perhaps, if you know of a user group for your kind of computer that does non-profit mail-order software distribution, please submit Kermit to them and let us know.

Have Kermit, Will Travel

Frank da Cruz and I have developed a series of Kermit seminars aimed for audiences ranging from novice to experienced. People in the New York City area may attend our courses at Columbia University's Division of Special Programs. Organizations outside of New York as well as local companies who would prefer in-house training might be interested to know that we have recently begun to take Kermit "on the road".

In November, 1987, Frank and I were invited to Tokyo to teach a Kermit course and give a Kermit presentation at Japan DECUS (Digital Equipment Corporation User Society). The course, *Fast-Paced Introduction to Kermit*, was conducted in English with Japanese translation. It covered the basics of terminal emulation and file transfer between MS-DOS PCs and

VAX/VMS, with a brief introduction to RS-232 asynchronous data communication and modems, and techniques for troubleshooting.

The presentation, *Kermit: Current Status and Future Directions*, was similar to the one prepared for Nashville DECUS in April 1987, except for the UN-style simultaneous translation and headphones, the guest speakers, and the news itself. We discussed Kermit's history, design and philosophy, gave a brief overview of the major new releases, and talked about Kermit performance issues and how they are addressed by compression, long packets, and sliding windows. Then Japan's "Kermit-san," Ken-Ichiro Murakami of NTT, spoke on special considerations for Japanese Kermits: implementations for Japanese computers, translation among the many computer representations for the different forms of Japanese text (Kana, Kanji, Romaji), Japanese-language Kermit documentation, and Kermit distribution in Japan. We were charmed by the generous hospitality of our hosts, and gratified by the high level of interest in, knowledge of, and support for Kermit in Japan. Many thanks to Kohichi Nishimoto, Administrator of Japan DECUS, for sponsoring the trip.

In March of this year, we were invited to Switzerland DECUS at St. Gallen to conduct a course, *Advanced Kermit: Use, Installation, and Support*, designed for computer professionals who are responsible for Kermit within their organizations. Unlike Japan, Switzerland has no common language, so Swiss DECUS is conducted in English. The all-day course was quite successful, and well-attended despite the blizzard outside. The topics ranged from tutorials in Kermit use and data communications to customization via script construction and key redefinition, illustrated using a MicroVAX 2000 and a Toshiba laptop connected to a projection screen. We are grateful to David Gürlet, Swiss DECUS Chairman, for the invitation and for his gracious hospitality. Course handouts included the new MS-DOS Kermit 2.30 and manual, plus MS-DOS Kermit documentation translated into German by Gisbert W. Selke of the Wissenschaftliches Institut der Ortskrankenkassen in Bonn, whom we had the opportunity to meet with in Ludwigsburg, West Germany.

Later in the month we met with the "Club Kermit" board of directors in Paris. Club Kermit is a nonprofit association, founded by Jean Dutertre, devoted to the distribution and promulgation of Kermit in France. Club Kermit may be contacted at 9 Av. Vergriand, 78600 Maisons Laffitte, France.

The entire Kermit Distribution collection was presented to both Japan and Switzerland DECUS, as well as to Club Kermit in France.

KERMIT DOCUMENTATION

Kermit manuals have been translated into Japanese, French, German, Italian, Hungarian and Portuguese, but so far we have only the Portuguese available online, plus some German help files. If you have machine-readable translations, please send them in!

Kermit User Guide, 7th Edition (May 1988). Detailed instructions for use of selected Kermit programs. New chapters for MS-DOS Kermit 2.30, UNIX Kermit 4E, IBM 370 Kermit 4.0 for VM/CMS and MVS/TSO, VAX/VMS Kermit 3.3, DEC-20 Kermit 4.2, PDP-11 Kermit 3.58, Apple II Kermit 3.81, Macintosh Kermit 0.9, and CP/M-80 Kermit 4.09. The printed version is typeset, with boldface, italics, underlining, etc, used to clarify the examples. The 7th edition of the User Guide is much thicker than the 6th, hence the higher price. Selected chapters are available separately: MS-DOS, Macintosh, UNIX, VMS.

Kermit, A File Transfer Protocol, by Frank da Cruz, Digital Press (1987). An introduction to, and comprehensive reference on, Kermit: its purpose, its use, its commands, its design. Complete with tables, illustrations, tutorials, appendices, a glossary, etc, as well as a troubleshooting guide and a complete specification of the Kermit protocol, with programming examples in the C language. Review clips:

- " . . . an unprecedented, self-contained book about a media-independent point-to-point 'filegram' protocol that is robust and adaptable to a wide variety of systems . . . an excellent reference source." –J. Klaczak, *ACM Computing Reviews*, March 1988
- " . . . addresses the needs of a full spectrum of users, from the computer semiliterate who has only used 'canned' programs, to the experienced user who merely wants to learn the Kermit command set, to the programmer who wants to implement Kermit on a new computer. [It is] a friendly book that puts you at ease with amusing and informative illustrations." –James Heliotis, *IEEE Computer*, December 1987
- " . . . highly recommended, not only for its KERMIT information, but also for the first 100 pages, which are one of the finest discussion of data communications anywhere." –Merv Adrian, "The Workstation Data Link", McGraw-Hill (1988)

Info-Kermit Digest, 1985-present. This is the electronic Kermit newsletter, issued approximately twice monthly, to thousands of readers on the international academic computer networks BITNET/EARN, CSNET, CCNET, etc, with contributions from Kermit users and developers all over the world. Printed versions are now available, paginated and indexed for easy reference. Included are announcements of new releases, discussions of proposed new features of the protocol, problems and solutions.

MAJOR NEW RELEASES

MS-DOS KERMIT 2.31

Short on the heels of version 2.30 of MS-DOS Kermit comes another new release, this time 2.31, also from Professor Joe R. Doupnik of Utah State University. But before talking about 2.31, let's correct a major omission in the announcement of 2.30 in the last issue: **Tektronix graphics terminal emulation**. IBM PC Kermit 2.30 and later includes emulation of the Tek 4010, plus some 4014 line drawing features, on most popular displays, including CGA, EGA, Hercules, Olivetti, AT&T, and even character-mode simulation on systems with no graphics boards at all!

Major new features for version 2.31 include expanded support for local area networks (see article on page 4), and new support for Kermit **file attribute packets**, so that file sizes, creation dates, and other information can be conveyed along with the file itself (page 6). And there's now a transaction log, so that you can come back after a long unattended file transfer session and find out what happened.

But the biggest change for 2.31 is in the **script and macro language**. It is now possible to define variables, to call macros with parameters, to construct loops, to perform conditional GOTOs, to reexamine previous INPUT strings, to delay operations until a predetermined time of day, and much more.

Thanks once again to Joe for his excellent work and generous spirit, and to the many beta testers for bug reports and fixes. MS-Kermit 2.31 is available on Tape A, prefix MS, and on diskette.

Since MS-DOS Kermit runs on so many different machines, not all versions are updated at once. IBM PCs and compatibles are done first, with other systems like Rainbow, Sanyo, HP, NEC, and Victor following later. As of press time, version 2.31 is available only for the IBM's and compatibles, the DEC Rainbow, the HP-150, and "generic MS-DOS."

MACINTOSH KERMIT 0.9(40)

Macintosh Kermit 0.9(40), by Paul Placeway of Ohio State University and Matthias Aebi of ECOFIN Research and Consulting, Ltd, Zürich, based on Columbia's original MacKermit, was released in May 1988. MacKermit 0.9(40) runs on all Macs except the 128K original, for which the previous release, 0.8(34), is still available.

As of 0.9, MacKermit has been translated into Apple MPW C, so that it can be edited, compiled, and built on the Macintosh itself. Previous versions had to be cross-compiled on a Unix system. MacKermit's new features include correct operation on the Mac II and SE; compatibility with all Macintosh keyboards; long packet support; a built-in, expanded key redefinition function; expanded server operation; multifile (folder) send; screen capture; transaction logging; Multifinder support; improved VT102 emulation, including the addition of smooth scrolling; and a new manual.

Thanks to everyone who tested different prereleases of this program since last October, and apologies to those of you who ordered MacKermit from us in the last several months and received a 0.9(36) beta-test copy without an up-to-date manual. And many thanks to Paul and Matthias for putting this release together and making it available. Both the new MacKermit and the old MacKermit (version 0.8(34) for the 128K Macintosh) are available on Tape B, prefix CK, and on Macintosh diskette.

KERMIT-370 4.0 FOR CMS & TSO

Portable IBM 370 Kermit was described by John Chandler of Harvard University in the last issue of the newsletter. Since then, this program has been released for both VM/CMS and MVS/TSO systems. It is a very advanced Kermit implementation, operating in both the linemode (3705) and full-screen 3270-emulation environments with a variety of protocol emulators, including the IBM Series/1, 4994, and 7171, as well as several non-IBM models.

Kermit-370 retires several redundant versions of TSO and CMS Kermit, in particular the venerable 1.0 for TSO, which had separate incarnations for each of several types of front ends. Kermit-370 is available on Tape B, with file prefix IK. It is distributed in IBM 370 assembly language source form, with complete assembly and installation instructions.

CP/M-80 KERMIT 4.09

This is the first new release of CP/M-80 Kermit since February 1985, when Charles Carvalho updated Columbia's original CP/M Kermit program to consolidate support for different systems and to add some new features. This process has been carried forward by Bertil Schou of Loughborough University in the UK, who has further modularized the program, improved its performance, added support for many new systems, and put in some new features too. Version 4.09 supports more than 60 different CP/M systems, ranging from the Kaypros and Osbornes, to various British micros, to the homemade models from the BYTE Magazine Circuit Cellar.

New features for this version include a selection of terminal emulations; standardized file transfer command syntax; command-line arguments; command and initialization files; TYPE, PRINT, and COPY commands; increased speed on DIRECTORY and wildcard SEND operations. The program is written in 8080 assembly language, and can be assembled using LASM or the Microsoft M80 assembler.

Many thanks to Bertil for this long-term effort, to Alan Phillips and Steve Jenkins at Lancaster University for sending this new version to us via transoceanic magnetic tape, and to the many others in the UK and elsewhere who contributed to the new release. CP/M-80 Kermit 4.09 is available on Tape A, prefix CP. As always, diskette volunteers and user-group diskette distribution are welcome!

CONTRIBUTIONS

MS-Kermit on Local Area Networks

Joe R. Doupnik
Utah State University

About one year ago Local Area Network support was incorporated in MS Kermit for IBM PCs, thanks to support and advice from AT&T. In contrast to the common use of LANs to connect workstations to network fileservers, Kermit uses the LAN to talk to other Kermits or to selected mainframe hosts; this is called peer-to-peer or any-to-any communications. The same MS Kermit program for IBM PC's which happily talks over a modem or a direct RS-232 connection can also use almost all popular LANs. These days when offices are wired for LANs but not RS-232, it is nice to know that Kermit can exploit the new high speed technology.

How does Kermit connect to a Local Area Network? Three kinds of network connections can be made. The first kind is through the industry standard NetBios interface (interrupt 5CH) which is supplied with virtually all current LAN systems. The Kermit command

```
SET PORT NET host-name
```

selects the LAN's NetBios channel, rather than the familiar serial port, and attempts to connect to the named host system. That is the only Kermit command which distinguishes network from serial port operations; everything else is exactly the same. All the LAN-specific work is carefully hidden behind the scenes and operates automatically. A BREAK signal can even be sent across STARLAN and Ungermann-Bass systems.

NetBios systems use names for each station and each active Kermit makes a unique name for itself. Connections to other Kermits are done by name. This technique works well even on LAN systems such as Novell Netware which do not otherwise really use

NetBios names.

The second connection method is to send and receive through a vendor-supplied intercept package which masquerades as a serial port. Many LANs and X.25 systems provide such packages. The Kermit command

```
SET PORT BIOSn (n = 1..4)
```

tells Kermit to use the system Bios so that the interceptor can capture traffic. Commonly, one "logs on" to the network package by behaving as a terminal (Kermit's Connect mode) and telling the interceptor the name of remote host, terminal type, and so forth according to the rules of the LAN system. Thereafter, Kermit operations are the same as using a modem or wired line. *(BIOSn support requires 2.31.)*

A third connection method is the Ungermann-Bass NetOne LAN system through UB's specialized NETCI (interrupt 6BH) LAN calls. The Kermit command

```
SET PORT UB-NET1
```

starts such a connection. Further "logon" commands are issued from Kermit's terminal emulator. UB support was provided by Rene Rehmann in Switzerland, and Henrik Levkowetz in Sweden. This kind of connection directs traffic to an outgoing bridge which simulates ordinary asynchronous character terminal activity to modems and other computers. *(NetOne support is available only in Kermit version 2.31.)*

What can MS Kermit do across a LAN? First, it works fast! As fast as the computers at each end can execute Kermit, roughly 40,000+ bits per second between standard AT's.

Second, Kermit can talk to other Kermits on the network as a Client (requestor) Kermit to a Server Kermit. Many such pairings can be active simultaneously without disturbing normal LAN operations. Each pair uses the NetBios unique names to make a formal connection, or "virtual circuit." Thus one or more Kermits can be named Servers and provide file exchanges between cooperating stations, at the whim of individual users. A Kermit Server (using the NetBios connection) left running all day can attend to many clients, one at a time of course. Since Kermit is not a Terminate-Stay-Resident program, workstation memory is freed when communications are finished.

Third, Kermit can act as a decent terminal (VT102, VT52, Heath-19 for text, and Tektronix 4010+ for both text and color graphics) for connections to mainframes and X.25 systems joined to the LAN; this seems to be one of the most popular options. Of course, where terminal traffic can flow usually one may find another Kermit to support file transfers.

What MS-Kermit cannot do: Well, it does not know how to converse with the LAN fileserver even though the network's pseudo disk drives work fine with Kermit. The reason is filesevers talk arcane proprietary languages across the LAN. Similarly, it does not know about other specialized operations associated with LAN servers, such as asynchronous modem servers, unless the LAN vendor supplies software replacing the normal serial port driver on the workstation (some X.25 systems and UB do this).

Kermit cannot act as a bridge to relay traffic from one source, say the LAN workstation, to another destination such as an attached modem. Kermit says "Packets stop here," to borrow from Harry Truman.

What is needed to use MS Kermit across a LAN? One each: MS-Kermit version 2.30 or later for IBM PCs (2.31 for BIOS*n* or NetOne support), a LAN of arbitrary size and manufacture, a matching LAN adapter board for the PC set to LAN adapter board number zero. Mix well, enjoy.

Dynamic Packet Size Control

John Chandler
Harvard University

Kermit-370 release 4.0 (for CMS and TSO and, potentially, other operating systems on IBM/370-type mainframes) has introduced a new feature with importance to the transmission of long packets, namely, dynamic tailoring of the packet size to noise conditions on the communication line. The implementation in release 4.0 is a greatly simplified version of a more detailed model that will be sketched out here. In essence, the model rests on a few simple assumptions:

1. Transmission errors have a Poisson distribution, i.e., occur randomly and not too frequently.
2. The effective speed is limited by both the transmission line data rate and the CPU processing speed (both of which add delays proportional to the number of bytes exchanged).
3. Each exchange of a packet and its acknowledgment is accompanied by a fixed communication dead time, which may include any combination of computer time sharing waits, line turnaround, satellite transmission delay, or other fixed overhead.
4. Noise between packets has no ill effects. Thus, errors occur effectively only in data packets, since those are much longer than the acknowledgments.
5. Each transmission error corrupts, but does not destroy, a packet.

With these assumptions, it is relatively straightforward to compute the throughput of a Kermit transmission in data bytes per second, given the relevant quantities:

- B, the block check type;
- P, the packet size (including the SOH and EOL);
- V_C, the computer encoding/decoding speed (serial, both computers);
- V_L, the communication line speed;
- X, the packet overhead (i.e., the number of non-data bytes in each packet);
- T_D, the packet dead time; and
- T_E, the average time between errors.

In these terms, the throughput is

$$\frac{(P-X) \times \exp(-P / (V_L \times T_E))}{(P+5+B) \times (1/V_L+1/V_C) + T_D}$$

A plot of throughput versus packet size for this model has a characteristic shape, rising sharply at small P, then leveling off and finally declining gently at large P. Since the throughput goes through a maximum, there is an optimum packet size, which can be found analytically from the expression for throughput. Indeed, if the average time between errors is reasonably long, i.e., if the mean number of characters transmitted between errors is much greater than 20, and if T_D is much less than T_E, the optimum packet size can be approximated by

$$\mathrm{sqrt}((X+5+B+D) \times V_L \times T_E)$$

where $D = T_D / (1/V_L + 1/V_C)$ is the number of bytes that could have been transmitted during the unavailable dead time.

In order to use this model, Kermit must have the theoretical quantities in terms of observable ones. For example, $V_L \times T_E$ is, in the long run, equal to the total number of bytes transmitted (both ways) divided by the number of errors, i.e., by the number of packets that must be resent. In short, Kermit can keep a running tally of bytes, packets, and retries during a transmission and from them compute the optimum packet size dynamically. If T_D is negligible, then Kermit needs only a total of bytes sent (and received) and a count of bad packets. Otherwise, Kermit must also have available the line speed V_L plus a count of packets exchanged and the elapsed time. From the tallied quantities, b, the total number of bytes exchanged; e, the number of error retries; p, the number of packet exchanges; and t, the elapsed time, a Kermit program can obtain the approximate optimum packet size in either of two ways. Assuming negligible dead time gives

$$\mathrm{sqrt}((X+5+B) \times b/e)$$

which underestimates the optimum, and making allowance for large dead time gives

$$\mathrm{sqrt}((b/p - (X+5+B)) \times (V_L \times t - b) / e)$$

which overestimates by virtue of assuming that V_C far exceeds V_L.

Kermit-370 has an implementation of the first expression, with the handy value 16 substituted for X+5+B (which would be 15 for B=1 and 19 for B=3). It is clear from the two expressions that the optimum cannot be calculated if there are no transmission errors; the optimum length would be infinite, and the packet size can be just what the other Kermit requested. When the long-packet protocol is in effect, Kermit-370 checks after every 20 packets during a SEND to see if there have been any errors and, if so, computes the optimum packet length based on the formula and uses that optimum size as an upper bound on subsequent packets. Simulations (with V_L at 120 bytes/sec) have shown that this method, while typically forcing the packet sizes to be smaller than optimal, achieves at least 94% of the maximum possible throughput as long as T_D is not significantly greater than one second and achieves essentially 100% when T_D is less than half a second. At higher speeds, the dead time is correspondingly more critical.

There are three main drawbacks to this simplified implementation. First, it does poorly with extreme T_D (but that could be remedied by adding an informational SET SPEED to Kermit). Second, it responds slowly to changes in the noise environment (but that could be improved by replacing the cumulative tallies with shorter-term subtotals). Third, it fails insofar as assumptions 4 or 5 do. For example, if errors caused packets to disappear, thereby triggering timeouts as well as retransmissions, a quite different model would be needed.

Kermit File Attributes

Frank da Cruz
Columbia University, New York City

The basic Kermit protocol allows one computer to send files to another, so that each file arrives at its destination correctly and completely, under its own name. But there are times when file transfer could be more useful if the receiving computer knew a little bit more about the file: its type (text or binary), its size, creation date, protection, or record format. These characteristics are called "file attributes".

Users of MS-DOS Kermit, for example, have long been able to watch the comforting "Percent Done" display when sending files, but when receiving them this information is missing. If the sending Kermit was able to tell the receiver in advance how big the file was, then not only could the receiver display the percent done, but it could also check whether there was enough space available for the file before agreeing to accept it, saving needless aggravation when disks fill up. Certain systems even require preallocated disk space before a file can be sent.

The Kermit protocol defines a way for Kermit programs to convey file attribute information along with the files they send. This is called the Attribute packet. One or more A-packets can be sent following Kermit's File header packet, and preceding the file data, provided both Kermit programs agree to do this during the feature negotiation phase. If they don't agree, files are transferred in the normal way, without attribute information.

The first implementation of the file attribute mechanism was done by Brian Nelson of the University of Toledo, Ohio, for PDP-11 Kermit in April 1984. This was necessary because transferring the name and contents of a file alone was not sufficient for the PDP-11's sometimes complex file and record structure. By using Kermit's "system-dependent" file attribute field, PDP-11 Kermit is able to transmit a file's entire directory entry along with the file, so that when the receiver is a PDP-11, any file can be stored in exactly the same way as on the originating system.

Kermit defines a series of "generic" file attributes, including size, date, type, and system of origin to allow the transferred file to reflect the original as much as possible. One attribute of particular interest is "disposition" – what to do with the file after it arrives. Normally, Kermit simply stores it on the current disk. The disposition attribute allows the sender to tell the receiver to print it instead, or to send it as mail to a specified user, or to submit it as a batch job, or to run it as a program, or to archive it. MS-DOS Kermit 2.31, for instance, includes a MAIL command, which allows the PC user to transfer a file to a Kermit server for forwarding to the given user as mail (unfortunately, there are no Kermits yet that can respond to this command, but the ability to mail should appear in the next release of C-Kermit).

Kermit file attributes are described in Chapter 12 of the Kermit book. Additional attributes are expected to be defined in the future, for instance to allow for representation of text files during data transfer in other than 7-bit US ASCII format according to evolving ISO 8859 standards for international character sets.

Like other advanced options, file attribute capability is gradually finding its way into Kermit programs. Those that support it today include MS-DOS Kermit 2.31 (A:MS), IBM PC Turbo Pascal Kermit 1.1a (A:TP4), Kermit-370 4.0 for VM/CMS and MVS/TSO (B:IK), PDP-11 Kermit (B:K11), as well as some of the commercial Kermit implementations. Attribute support is planned for the next release of C-Kermit.

KERMIT ORDERING INFORMATION #____________

Allow 2-4 weeks for delivery. Kermit tapes include program source code and online documentation. All tapes are half-inch, 2400-foot, 9-track, 1600bpi, odd parity, and available in the following formats:

ANSI:	ANSI labeled ASCII, format D (variable length records, VMS COPY), blocksize 8192
TAR:	UNIX TAR format (written on a VAX with 4.2BSD or Ultrix-32), blocksize 10240
OS:	IBM OS standard labeled EBCDIC, format VB (variable length records), blocksize 8192
CMS:	IBM VM/CMS VMFPLC2 format (unlabeled)
DEC-10:	DECsystem-10 Backup/Interchange format (unlabeled) (LAST CHANCE!)
DEC-20:	DECSYSTEM-20 DUMPER format (unlabeled) (LAST CHANCE!)

TAPE CONTENTS:

Tape A: The more popular microcomputer (PC, workstation) Kermit implementations: Apple II, Atari, Commodore, CP/M, Intel, MS-DOS, TRS-80, etc.

Tape B: The more popular mainframe and minicomputer Kermit implementations: UNIX, VAX/VMS, IBM mainframe, DEC PDP-11, DEC-10/20, etc. All C-Kermit versions including micros (Commodore Amiga and Apple Macintosh).

Tape C: Less popular microcomputer Kermit implementations (overflow from Tape A): Acorn, Alpha Micro, Apollo, OS-9, Concurrent CP/M-86, Convergent, LISP machines, HP workstations, Perq, RML, UCSD p-System, etc.

Tape D: Less popular mainframe and minicomputer Kermit implementations (overflow from Tape B): Burroughs, Data General, CDC, Cray, Computervision, Honeywell, GEC, Gould, Harris, HP minis, DEC PDP-8, MAI Basic Four, MODCOMP, Perkin-Elmer, PICK, Prime, Sperry, Tandem, TI, ICL, etc.

Tape E: Machine readable copies of the Kermit manuals, including text formatter source. Archives of Kermit network mail digest, other large documents. Articles. Discussions of specific topics, like IBM mainframes and international character sets. Translations into other languages, like Portuguese. Tape utilites.

9-TRACK MAGNETIC TAPES. All Kermit source is included, and each tape order comes with a printed copy of the Kermit User Guide and Kermit Protocol Manual. Price: $100 PER TAPE. Check each desired Kermit Distribution Tape:

Tape Format:	ANSI	TAR	OS	CMS	DEC-10	DEC-20
Tape A (popular micros):	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
Tape B (popular mainframes):	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
Tape C (other micros):	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
Tape D (other mainframes):	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
Tape E (documentation):	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]

[ ] PRIME Computers: Specify ANSI; Check here for free Fortran ANSI-tape-reader program listing.

TAPE SUBTOTAL (number of tapes times $100) . . . . $____________

MicroVAX TK50 tape cartridges, $250 EACH, manuals included:
[ ] *Contents of Tapes A, B, and E, VMS BACKUP Format* . . . . $____________
[ ] *Contents of Tapes A, B, and E, Ultrix tar Format* . . . . $____________
[ ] *Contents of Tapes C, D, and E, VMS BACKUP Format* . . . . $____________
[ ] *Contents of Tapes C, D, and E, Ultrix tar Format* . . . . $____________

Kermit programs on diskette (not copy-protected), no source code, $20 EACH, manual included:
[ ] *IBM PC, XT, and AT; PC-DOS* (5.25 inch 360KB DS diskette) . . . . $____________
[ ] *IBM PS/2 or PC Convertible; PC-DOS* (3.5 inch 720KB DS diskette) . . . . $____________
[ ] *Apple Macintosh* (3.5 inch SS diskette) . . . . $____________
[ ] *DEC Rainbow; CP/M-86* (RX50) . . . . $____________
[ ] *DEC Rainbow; MS-DOS* (RX50) . . . . $____________
[ ] *HP-150; MS-DOS* (3.5 inch diskette) . . . . $____________
[ ] *DEC MicroVAX; VMS* (RX50) . . . . $____________
[ ] *DEC MicroVAX; Ultrix* (RX50) . . . . $____________

SIDE 1 SUBTOTAL (please complete side 2 also) . . . . $____________

SUBTOTAL FROM SIDE 1 . . . . $____________

Kermit source code on IBM PC 5.25 inch 360KB diskettes, manual included:
[ ] MS-DOS IBM PC Kermit MASM Sources, 4 disks, $50 . . . . $____________
[ ] C-Kermit UNIX C-Language Sources, 2 disks, $30 . . . . $____________

Printed documents, enter quantity:
[ ] Book: *Kermit, A File Transfer Protocol*, 1987 ($26) . . . . $____________
[ ] Manual: *Kermit User Guide*, 1988 ($20) . . . . $____________
[ ] Manual: *Kermit Protocol Manual*, 1986 ($10) . . . . $____________
[ ] Separate chapters from the Kermit User Guide ($10 each):
  [ ] MS-DOS, [ ] UNIX, [ ] VMS, [ ] Macintosh . . . . $____________
[ ] *BYTE* Magazine article manuscript, 1984 ($5) . . . . $____________
[ ] *Data Communications* Magazine article manuscript, 1987 ($5) . . . . $____________
[ ] Info-Kermit Digest Volumes, paginated & indexed ($15 each):
  [ ] V3 1985, [ ] V4 '86, [ ] V5 '86 . . . . $____________
  [ ] V6 1987, [ ] V7 '88 . . . . $____________

[ ] ORDER PROCESSING FEE: If you can NOT prepay with a check, please include **BOTH**:

1. A $100.00 Order Processing (Billing) Fee: . . . . $____________
2. **AND** a Purchase Order (terms net 30, FOB origin):

Purchase Order Number: ________________________________
(not required if you prepay by check.)

[ ] OVERSEAS SHIPPING: Outside North America, add $25.00 for postage . . . . $____________
Or provide your [ ] Federal Express or [ ] DHL account number:

______________________________________(*Warning*: postal shipment may be slow)

[ ] USA RUSH ORDERS (Priority handling, Federal Express), add $30.00 . . . . $____________

[ ] VOLUNTARY TAX-DEDUCTIBLE CONTRIBUTION . . . . $____________

GRAND TOTAL, SIDES 1 AND 2 (*Do Not Add Sales Tax*): . . . . $____________ (@)

IMPORTANT: WRITE SHIPPING ADDRESS HERE, *exactly as it should appear on mailing label*. Please don't use a Post Office Box number, because UPS or other delivery services cannot deliver to PO Boxes.

Name:______________________________________ Organization:____

Address: ____

City:_________________________State or Province:______________Zip or Postal Code:______________________________

Country:____Phone: ____

MAIL COMPLETED ORDER FORM TO:

Kermit Distribution
Columbia University Center for Computing Activities
612 West 115th Street
New York, NY 10025 (USA)

MAKE CHECKS IN US DOLLARS PAYABLE TO: **Columbia University Center for Computing Activities**

Summary of Recent Releases

Here is a concise list of Kermit releases since our last issue in November 1987. The notation (A:XYZ) means the version is available on tape A, as files whose names start with XYZ; "diskettes" mean the item is also available from Columbia Kermit Distribution on diskette (see order form). Volunteers listed when available.

Micros and PCs:

- MS-DOS Kermit 2.31 for the IBM PC and PS2 family and compatibles, the DEC Rainbow, the HP-150, and generic MS-DOS, written in Microsoft Macro Assembler (MASM) by Joe Doupnik, Utah State University, June 1988 (A:MS, diskettes). See articles on pages 3 and 4.
- MS-DOS Kermit 2.30 for NEC APCIII, MASM, Bob Goeke, MIT Center for Space Research, Mar 88 (A:MS). Also on 5.25" diskette from Robert Goeke, MIT Center for Space Research, Room 37-567, 77 Massachusetts Ave, Cambridge, MA 02139. Send 1 disk for executable; 4 disks for source code, and a stamped return mailer.
- MS-DOS Kermit 2.30 for the Sanyo MBC 555, MASM, Bob Babcock, Harvard/Smithsonian Center for Astrophysics, May 88 (A:MS). Also available on 5.25" diskette from Bob Babcock, Sanyo New England Users Club Librarian, 4 Reeves Road, Bedford, MA 01730, USA. Cost: $5 for executable and documentation, $10 for same plus source code on DS disks, $6/12 for SS disks. Foreign orders: Pay in US funds, add $3 shipping.
- MS-DOS Kermit 2.30 for the ACT Apricot, MASM, Dick Carlton, The Open University, UK, May 88 (A:MS).
- Intel iRMX-86 and iRMX-286 Kermit 2.30, based on MS-DOS Kermit 2.30, Jack Bryans, California State University at Long Beach, May 88 (A:MS). Provides most of the functions of IBM PC MS-Kermit. Also on 5.25" diskette for $10 from CSULB University Bookstore Computer Shop, 6049 E. 7th Street, Long Beach, CA 90840 (specify Disk 1).
- MS-DOS QK-Kermit Version 3.0 in Turbo Pascal Version 4 for the IBM PC family, with Tektronix emulation, Victor Lee, Queens University, Kingston, Ontario, Apr 88 (A:QK). Improved graphics, support for a variety of graphics adapters. Also available on diskette for $10 (Canadian or US, incl. manual) from Victor Lee, Computer and Communications Services, Dupuis Hall, Queen's University, Kingston, Ontario, CANADA K7L 3N6 (USA & Canada only).
- MS-DOS Kermit 1.1a for the IBM PC family and compatibles, Turbo Pascal, Terje Mathisen, Norsk Hydro Data, Norway, Apr 88 (A:TP4). Sliding windows, long packets, attributes. Rudimentary dumb terminal emulation. This version came to us indirectly; we've had no contact with the author.
- Macintosh Kermit 0.9(40) from Paul Placeway and Matthias Aebi, MPW C, Jun 88 (B:CK, diskette). See article on page 3.
- Apple II DOS and ProDOS Kermit-65 Version 3.81, 6502 Assembler, Ted Medin, Feb 88 (A:APP). New features include wildcard send, improved VT100 emulation, new manual. Also available on 5.25" diskette for $10 from Peter Trei, 15 Sickles Street, New York, NY 10040 (USA only).
- Commodore 64/128 Kermit 2.1(68), 6502 Assembler, Ray Moody, Purdue University, Apr 88, (A:C64). New features include limited Tektronix 4010 emulation and improved VT100 emulation. Also available on diskette for $5 ($10 with manual) from Dr. Evil Laboratories, P.O. Box 190, St. Paul, IN 47272 (USA only, Indiana residents add 5% sales tax).
- CP/M-80 Kermit Version 4.09, LASM, Bertil Schou, Loughborough University, UK, 18 May 88 (A:CP). See article, p.3.

Minis, Mainframes, and Supercomputers:

- IBM Mainframe Kermit 4.0 for VM/CMS, 370 Assembler, John Chandler, Harvard/Smithsonian Center for Astrophysics, Dec 87 (B:IK). See article on page 3.
- IBM Mainframe Kermit 4.0 for MVS/TSO, 370 Assembler, John Chandler, *loc. cit.*, Mar 88 (B:IK).
- VAX/VMS Kermit-32 3.3.117, the Stevens Institute of Technology Bliss-language version, modified by Jonathan Welch of Amherst College, MA, to fix several bugs and add some minor features, Jun 88 (B:VMS).
- C-Kermit 4E(070) for UNIX, VMS, etc, in C, Frank da Cruz, Columbia University, New York, Jan 88 (B:CK). Minor fixes and improvements to 4E(067), the first long-packet C-Kermit release.
- C-Kermit 4E(070) adapted to OS-9/68K, Bob Larson, Feb 88 (B:CK).
- C-Kermit 4D(061) adapted to Andrew Tanenbaum's MINIX, Adrian Godwin, Bedford, UK, Feb 88 (B:MX).
- CDC Cyber NOS Kermit 3.4, Fortran, Steve Roseman, LeHigh University, and Olaf Pors, University of Virginia, May 88 (D:CD3). Wildcards, compression, long packets, init & take files, support for CDCNET, bug fixes, many new commands.
- DECSYSTEM-20 Kermit 4.2(262), Assembler, Frank da Cruz, Jan 88 (B:K20). Minor improvements.
- Honeywell CP-6 Kermit 1.00, PL/6, Lee Hallin, Honeywell Bull, Los Angeles, CA, Jan 88 (D:HC6). Improved performance, bug fixes, some new features.
- Harris-100 Kermit 1.06, Fortran-77, Skip Russel, Washington University, St. Louis, MO, Mar 88 (D:H1). Bug fixes.
- MAI Basic Four Kermit in Business Basic, E.V. Wastrodowski, Sphere Holdings Ltd, Laguna Hills, CA, Apr 88 (D:MBF). First Kermit release for this system.

THIS IS REALLY THE BACK PAGE, WHERE THE MAILING LABEL GOES...