news@ftp

The FTP Software® Newsletter

Volume II, Number 1 May 1989

Inside…

We’ve moved!

FTP Software moved from our old offices in Cambridge to new corporate headquarters in Wakefield on February 25. Our new address for both shipping and correspondence is:

FTP Software
26 Princess Street
Wakefield, MA 01880-3004
Phone: (617) 246-0900
Fax: (617) 246-0901
Telex: 981970

We are now occupying all of the third floor and half of the second floor of an office building in downtown Wakefield (shown at right). This allows us the room we need to expand both our testing facilities and our staff.

We’ve hired many new people during the past year to try to satisfy the needs of our customers and distributors, and, as a result, we grew out of our old location. The staff that we’ve hired over the past year includes people in technical support, documentation, sales and production.

Open House

To celebrate our move and to show our customers and affiliates our new offices, we will be having an open house on June 1, 1989 from 2:00 to 7:00 PM. Everyone is invited to stop by and see our new location and meet the members of our staff or catch up with old acquaintances.

news@ftp

Volume II, Number 1 ¥ May 1989

Published by FTP Software, Inc.
Editor: Nancy Connor.
Communications pertaining to this newsletter should be directed to:

news@ftp
FTP Software, Inc.
26 Princess Street
Wakefield, MA 01880-3004
Phone: (617) 246-0900
FAX: (617) 246-0901
Telex: 981970
Answerback: FTP SFTWRE UD
Electronic mail: news@ftp.com

FTP Software welcomes suggestions and contributions from readers. Submissions for the next issue of the newsletter must be received no later than August 1, 1989.

PC/TCP® Plus and InterDrive Start Shipping!

We have just released InterDrive, our newest product! InterDrive is our implementation of NFS™ for the IBM® Personal Computer®. We are very excited about the release, because customers have asked for NFS more than any other enhancement to PC/TCP.

With InterDrive users have access to the storage capacity of NFS servers. Not only does this free up space locally, but because files are stored centrally, many people can access commonly referenced files.

InterDrive is available as an add-on to PC/TCP, version 2.02 or greater, for $100. PC/TCP Plus is a new package which contains PC/TCP version 2.03 and InterDrive, and sells for $490. Upgrades for older versions of PC/TCP are also available. Site license pricing is available for all of these products. Ask for the InterDrive data sheet for more details.

Several OEMs Add NetWare™ Compatibility

The number of FTP Software OEMs supporting our Packet Driver Specification has swelled in recent months. Each of the new OEMs has written a Packet Driver for PC/TCP that provides full compatibility with NetWare. The new vendors are Acer™, D-Link®, Gateway Communications and IMC Networks. They join the ranks of BICC Data Networks, Proteon, Schneider & Koch and Univation, who have been providing PC/TCP-NetWare compatibility for up to two years.

PC/TCP-NetWare compatibility is a major benefit for NetWare customers, because they can alternate between NetWare and PC/TCP commands without rebooting. They can even use both protocols at the same time! For example, they can use FTP to transfer a file from a TCP/IP host to a NetWare server.

Vendors or customers who are interested in writing a Packet Driver for a particular card simply need to obtain a copy of PC-210 and the Packet Driver Specification. Included with PC-210 are a set of public domain packet drivers, some of which include source, that can be used as the basis for your new Packet Driver. To obtain NetWare compatibility, the NetWare driver also needs to be modified to conform to the Packet Driver Specification.

Interrupt 14 Telnet Upcoming

Users of many commercially available terminal emulation packages will be glad to know that the next release of PC/TCP (due out in several months) will contain an interrupt 14 Telnet. Many popular emulators will then run unmodified over PC/TCP.

Interrupt 14-compatible emulators are available for a wide variety of terminals, including DEC, HP and Tektronix® terminals. Among the vendors who supply interrupt 14 emulators are Diversified Computer Systems, KEA Systems, Persoft, Polygon, Softronics® and Walker Richer & Quinn. For users who don't want to wait for the next version of PC/TCP, the pre-release version of the interrupt 14 Telnet is available for $40 per copy (for PC/TCP versions 2.02 and greater).

Three vendors also offer emulation packages for the current version of PC/TCP (2.03). Walker Richer & Quinn offers a package called Telnet Manager, which allows all of their Reflection® Plus series of DEC, HP and Tektronix emulators to run over PC/TCP Plus. Diversified Computer Systems series of DEC and Tektronix emulators is also compatible with PC/TCP, and Grafpoint™ offers several compatible Tektronix emulators.

Using Token Ring Networks

By Benjamin Levy

Writing drivers for the IEEE LAN standard 802.5 Token Ring network which is also known as 4 megabit Token Ring, IBM Token Ring and ProNET-4® networks has taught me a lot about how the Token Ring works and what its problems are. In the following paragraphs I will try to share some of that knowledge with you.

There are two ways for network applications to communicate with the Token Ring cards. Either they can talk directly to the card (handle interrupts from the board, transfer data to and from the board, etc.) or they can talk to an intermediate software interface, which handles all the details of how the board works. Many network applications use the latter method. IBM created a device independent interface for Token Ring boards called the Adapter Support Interface, or ASI. This enables the end user to use network applications which were designed to work on IBM Token Ring boards with Token Ring boards made by other manufacturers.

IBM makes two different versions of the ASI. The older version is called *tokreui.com* and it can be loaded into memory at any time. The new version of IBM s ASI is called the Lan Support Program. The Lan Support Program is intended to be used with DOS™ 3.3 and higher. It will also work on a wider variety of IBM computers, including PS/2s. The Lan Support Program is made up of two program modules which are called *dxmc0mod.sys* and *dxma0mod.sys*. The modules are loaded into memory by *config.sys.*

Many of the other Token Ring manufacturers have ASIs for their Token Ring boards. Proteon has an equivalent of IBM s *tokreui.com* called *pro4eui.exe*, Sytek has *asi.exe* and 3Com™ has *3comreui.exe*.

To use the new ASI with non-IBM Token Ring boards (i.e. Proteon, 3Com, and others) you will need DLC ROMs on the Token Ring board. If the board has a DLC ROM version less than 2.08 you may have to get a new set from the dealer or manufacturer. The DLC ROMs can usually be found in sockets on the left hand side of the board and will usually have DLC written on them and have a TI, IBM or Sytek copyright.

NCR, Proteon, Sytek, Western Digital, and 3Com make drivers that can work with IBM s

Lan Support Program by replacing IBM s *dxmc0mod.sys* with their own version of the driver. Note that you must also purchase the Lan Support Program from IBM or a dealer even though you are using a Token Ring board from a different manufacturer.

IBM makes three Token Ring cards: the Token Ring Network PC Adapter, the Token Ring Network PC Adapter II, and the Token Ring Network Adapter/A for Micro Channel™ Adapters. The PC Adapter II can be distinguished from the PC Adapter by the extra memory on the PC Adapter II board.

Proteon makes four boards for their ProNET-4, which is functionally equivalent to the IBM Token Ring. The p1340 is an 8 bit board for either XT™ or AT® bus PCs, the p1344 and p1345 are 16 bit boards for AT bus PCs, the p1345 is a new version of the p1344 which has been modified to work with PC-AT clones that are not 100 percent compatible with IBM s PC/AT, and the p1840 is a Micro Channel Adapter board for the Personal System/2™.

3Com makes the TokenLink™ (part number 3C603) and TokenLink Plus (part number 3C605). Western Digital makes the TokenCard™ (part number WD8005-TR) and TokenCard WS™ (part number WD8005-TRWS). All of these boards work in PC, XT, or AT bus PCs.

Other Token Ring boards I ve heard of but haven t tried, are a Token Ring board made by Gateway Communications, NCR s PC Token-Ring Connection, Sytek s 6140 Token Ring card (which is also resold by other OEMs), and a board made by Ungermann-Bass.

How the Token Ring works

The IBM and Proteon Token Rings pass data at a rate of 4 Megabits per second. Each board in the Token Ring plugs in to a wire center. But the board doesn’t actually become part of the ring until the board is initialized by the software and is told to enter the ring. Boards in the ring take turns talking, and this is regulated by a token. In each ring there is one token, which gets passed from one board in the ring to the next. If a board has a packet to send, it must wait until it gets the token to put the packet on to the ring. The packet will then be passed from one board to the next, until it reaches the recipient. When the recipient receives the packet it will set a flag in the packet, marking the packet as received. Then it puts the packet back in the ring. When the packet gets back to the sender, the sender will be able to check if the packet has been read by the recipient.

The first board to enter the ring becomes the active monitor. The active monitor constantly watches the traffic on the ring, checking for a missing or duplicate token. If either occurs, it will flush the ring, then it will put a new token on the ring. The active monitor also keeps track of the boards in the ring, and it will inform the other boards in the ring if it notices that one of the boards is malfunctioning. The other boards in the ring are standby monitors, waiting to take over if they notice that the active monitor has left the ring, or if it is having problems. All of the monitoring is done automatically by the Token Ring board, and does not interfere with the other operations of the PC.

The first two protocol layers of the packets on the Token Ring are the Physical or MAC (Media Access Control) Layer, which is described by IEEE standard 802.5, and the Data Link Layer, which is described by IEEE standard 802.2. The MAC header contains a two byte field which indicates the packet type, the source and destination hardware addresses, and the Routing Information Field (RIF). The 802.2 header is made up of the LLC (Logical Link Control) layer and the SNAP (Sub-Network Access Protocol). For the purposes of encapsulating IP over 802.2, the values for the LLC and SNAP headers are constant, except the value of the Ether Type field in the SNAP header. The next layer is the IP layer and is the same as the IP layer over the Ethernet (or any other network type).

The way RIFs are used is as follows: When you want to open a connection to another machine, the PC sends out an address resolution protocol (ARP) packet asking for the hardware address of the destination machine. Each bridge on the Token Ring will add its two byte address to the RIF as it passes the packet across the bridge to the next Token Ring. When it finally gets to the destination machine, the RIF will contain the path needed for sending the reply. The destination machine will then send a reply back to the original sender which contains the RIF with the path in it and its hardware address.

PC/TCP and LANWatch®

FTP offers two software applications for the Token Ring. There is PC/TCP, which is a set of network applications for intercommunication, and LANWatch, which is a network analyzer program. A version of PC/TCP (part numbers PC-108 and PC-208) talks directly to Proteon s ProNET-4 boards, and another version (part numbers PC-119 and PC-219) uses the Adapter Support Interface. A version of LANWatch (part number LW-108) uses Proteon’s ProNET-4 boards. No version of LANWatch exists for the IBM Token Ring cards because of IBM security features built into those cards.

LANWatch on Proteon s ProNET-4 cards requires DLC ROMs (version 2.01 or higher). Some of Proteon s earlier boards didn’t come with the DLC ROMs needed for LANWatch. You can check for their presence by looking on the the left-hand side of the board. If there are two or three empty chip sockets on the board, then you’ll need to order the DLC chips from Proteon. Their part number is p5412-12 for the p1340 board and p5412A-12 for the p1344 and p1345 boards. The DLC chips also come with a disk containing *pro4eui.exe*.

All of the Token Ring boards that I have tried work with our PC/TCP software, provided that the DLC ROMs on the Token Ring boards are version 2.08 (or higher).

If you are using PC/TCP with an ASI, it is advisable to start PC/TCP first. This allows it to initialize the ASI and to open the Token Ring board before you use other manufacturer s network applications.

While testing the driver for PC-119 and PC-219, it was discovered that the ASI was causing problems by writing over memory randomly. To fix this problem, *watsr.exe* was created. *watsr.exe* is a simple terminate and stay resident program that sets aside 256 bytes of memory for the ASI to use, then passes the address of the work area to the PC/TCP programs so they can tell the ASI to use the work area provided to it.

Since the release of the 2.03 version of PC-208 and PC-219, and the release of the 1.16 patch level 5 version of PC-108 and PC-119, PC/TCP has been able to work with RIFs.

Problems

There are some problems that may occur when using Banyan s VINES™ file server (version 3.01) as a gateway for Proteon and 3Com Token Ring boards. Apparently the Banyan VINES file server was having problems with the manufacturer address part of the hardware addresses on the boards. Banyan is aware of this problem and they are in the process of fixing it. Until then, to get around the problem just tell the board to use an address that doesn t have the first bit of the third byte set. In both cases, the address being used by the board can be changed at the software level. The first three bytes of Proteon s hardware address are 40-00-c9, and the first three bytes of 3Com s address is 42-60-8c. If the address is changed, the addresses become 40-00-49 and

42-60-0c respectively.

Proteon:

```
 4    0    0    0    c    9
 0100 0000 0000 0000 1100 1001 ...
                     ^
                     offending bit
```

3Com:

```
4    2    6    0    8    c
0100 0010 0110 0000 1000 1100 ...
                    ^
                     again
```

The same addresses with the first bit of the third byte set to 0.

Proteon:

```
 4    0    0    0    4    9
 0100 0000 0000 0000 0100 1001 ...
```

3Com:

```
 4    2    6    0    0    c
 0100 0010 0110 0000 0000 1100 ...
```

Note that the Token Ring board will tell you if the cable is not attached to the board, but it won t tell you if the cable is not attached to the wire center. If you are having a problem with a Token Ring board that is neither sending nor receiving packets, try checking that the far end of the cable is attached to the wire center.

If you are using IBM s LAN Support Program with IBM s NetBIOS interface, you must tell *dxmt0mod.sys* to leave extra service access points (SAPs) for other programs to use (e.g. PC-119 or PC-219). This can be done by adding the option ES=1 or ES=2 to *dxmt0mod.sys* in *config.sys.* Read the documentation on the disk that contains the software for more information about IBM s LAN Support Program and its options.

If you are using FTP s NetBIOS program with PC-219, you do not need to load *dxmt0mod.sys* in *config.sys.*

Acknowledgements

I d like to thank Ellie Gittelman of Proteon, Inc. and Drew Perkins of Carnegie Mellon University for their assistance.

References

For a more detailed description of how the Token Ring works see the following sources:

IEEE Standard for Local Area Networks: Logical Link Control. ANSI/IEEE Standard 802.2, 1985. Produced by The Technical Committee Computer Communications of the IEEE Computer Society.

A Standard for the Transmission of IP Datagrams over IEEE 802 Networks. RFC-1042. Written by J. Postel and J. Reynolds.

TMS380 Adapter Chipset User s Guide. Produced by Texas Instruments

TMS380 Adapter Chipset User s Guide Supplement. Produced by Texas Instruments.

Token Ring PC Network Adapter Technical Reference. Printed by International Business Machines Corporation, 1986.

Recent Workstation-Oriented Enhancements to Telnet

by James VanBokkelen

While many modern personal computers and workstations have bit-mapped displays, most applications software still expects to communicate with an asynchronous terminal. Mechanisms like the BSD TERMCAP variable and VMS Screen Management allow applications to support many kinds of asynchronous terminals, and workstation windowing software typically includes one or more asynchronous terminal emulations. The transition from simple terminal-style user interfaces to more sophisticated schemes is evident in many places, but it will take a long time to complete. Meanwhile, two recent RFCs describe additions to the Telnet protocol which make things easier on the users stuck in the middle.

Negotiate About Window Size Option Defined

RFC 1075, by David Waitzmann of BBN, defines a mechanism to pass information about display window sizes and size changes from client to server (and to co-operating full-screen applications). A new Telnet option, Negotiate About Window Size (NAWS), is specified, and client and server are expected to negotiate it with WILL/WONT/DO/DONT like other options. If NAWS is agreed on, then the client can send Telnet Sub-Negotiation sequences to the server to indicate the initial display aperature dimensions (in characters), and further Sub-Negotiation sequences later if the dimensions change.

Two previously defined Telnet options, Negotiate Output Line Width and Negotiate Output Page Size, had never entered common use, and RFC 1073 comments that they were not adequate for this use, in any case. NAWS allows window dimensions up to 65536 x 65536, and establishes both dimensions in one operation. The older options were limited to 253 x 253, and required one operation per dimension changed. Since most window changes affect both dimensions, this might result in two successive screen updates being sent to the client.

Terminal Type Option Enhanced

RFC 1091, by James VanBokkelen of FTP Software, defines the existing Telnet Terminal Type (TTTYPE) option, to allow the Telnet server to review the client s list of supported terminal emulations, and select the one it likes best. Later, possibly at an application s request, another terminal type from the supported list can be selected.

As the option had previously been defined in RFC 930, the client indicated end-of-list by sending the last supported type a second time. Thus, a server could only see the list of types once, and because the most desirable type was listed first, most existing implementations simply settled for that one. Also, multiple emulation modes had not really been dealt with, and the exact point at which the mode changed was not specified.

The new specification builds on the old, while maintaining backwards compatibility with RFC 930. The option value remains the same, and old servers will receive the same end-of-list indication as before. However, if a new server sends one more request after receiving the end-of-list marker, a new client is expected to return to the top of its list. The mode change is defined as taking place when each new type is sent.

What the User Can Expect

When will these options show up in your environment? That depends on the energy of your vendors (or the authors and maintainers of any public domain software in use at your site). NAWS is most important where windowing software is widely used, so its first implementations have been in public-domain software packages maintained at CMU. TTTYPE is more important in business and non-Unix environments, where applications are more likely to be coded for use on only a few terminals. There are already a couple of forward-looking TCP/IP vendors with plans for supporting the extended Terminal Type option.

Sun Sponsors NFS Connectathon

Sun Microsystems sponsored an NFS Connectathon on February 18 through 22 in Santa Clara, CA. It consisted of a technical forum for testing NFS implementations from the various vendors, plus a number of sessions on topics relating to NFS.

The purpose of the Connectathon was to provide a friendly forum for NxN testing of the NFS products currently available or under development. NxN testing means that every implementation of the software in question tests against every other implementation that is present at the test, where N is the number of implementations.

There were around 70 vendors represented, and about 60 of them had NFS implementations of one sort or another. The other vendors were present in order to test their X-window software. Of all of the NFS implementations, only a few were PC products.

As always in these sessions, there was some initial confusion on the technical side of the Connectathon. Most of the problems were due to lack of information, for example, what addresses each of the vendors should use, what broadcast addresses were valid, and some technical questions about specific NFS implementations.

Sun Microsystems put together an NFS test suite which provided a fair level of functional testing. Unfortunately, their test suite was not available for DOS, so those venders who had DOS implementations got together a test suite from a combination of software from a few of the vendors.

While the test suite for DOS was being prepared, we started our own tests against the other vendors software. The testing procedure was very simple you walked over to one of the vendors and said, Hi, I d like to test against your server. Then they would tell you what to mount, and what your userid should be, and you d begin testing against them. At first, many of the implementations did not perform well, but as time went on most of them were in very good shape. By the time the Connectathon ended a large percentage of the vendors had tested against most of the other implementations.

In addition to the testing, there were sessions on various subjects. However, most of the true learning seemed to occur in the technical portion by running the test suites, watching the network, listening when other people discussed their problems, and talking to other people about their implementations. The people from FTP who attended the event learned a lot by their participation.

As may be expected, a spirit of cooperation prevailed during the event, allowing many vendors to work together to ensure interoperability. The vendors were willing to assume that their code might be as much at fault as the other side. We d like to extend a special vote of thanks to Sun Microsystems for sponsoring the event, and Carl Beame from Beame and Whiteside for the time he spent debugging code from other vendors.
