OPCUG The Ottawa PC Users' Group

Mailing Address - 3 Thatcher Street, Nepean, Ontario, K2G 1S6

JANUARY 1991 (613) 723-1329 (voice) (613) 723-7064 (BBS) VOL. 8, NO. 1

DOES dBASE IV 1.1 REALLY WORK?

By Christopher Studholme

Since Ashton-Tate's presentation at the November General meeting of the OPCUG, many people may be wondering if dBASE IV Version 1.1 works as well as the Company claims it does. I have had the opportunity to test the product, and I hope that I can provide some answers based on my hands-on experience.

The history of dBASE IV began with the database wars which have been heating up over the past several years. During most of this period, the Ashton-Tate people sat smugly behind their dBASE product line, dominating the market and waiting for new challengers.

The competition was not very strong at [illegible] but quickly caught up with numerous products such as Paradox, R:base, Oracle, Clipper, FoxPro and a host of no-name and smaller user-specific databases. Ashton-Tate found itself suddenly caught in a war for market share and under pressure from dedicated users for an improved product. Thus, we saw the introduction of dBASE IV 1.0.

After the fanfare died down and programmers started using the product, they found it contained serious problems which are too numerous to mention here.

Ashton-Tate had committed the ultimate marketing sin. They had produced and released a package without checking for flaws.

The resulting fall-out proved to be devastating. Not only had the Company damaged its reputation, it was also losing its share of the market.

Ashton-Tate reacted by producing a cleaned-up and improved version of dBASE IV called dBASE IV 1.1. Rumours and accusations once again swirled around. People I relied on for information advised [illegible]gainst using the product. Nonetheless, [illegible]cided to go ahead and check it out myself.

During an interview with Mr. Michael Burrows, Manager of Ashton-Tate for Ottawa, I found out how extensively 1.1 was tested. It was given for trial to over 100 employees within the Company. Also, over 60,000 automated tests were performed, and a beta-testing program was carried out at 2,700 sites around the world.

Armed with this knowledge, I tested the product on several types of computers to see if it performed equally well on all of them. It worked on all machines with 101 enhanced keyboards, but I could not get it to respond completely on full-sized 83 keyboards, such as those that come with Tandy personal computers.

Mr. Burrows assured me that anyone getting in touch with Ashton-Tate would receive the appropriate keyboard driver for their computer. My personal advice, however, is to stick with a 101 enhanced keyboard.

I found dBASE IV to be a lot more user-friendly than dBASE III+. All the functions that were on III+ are still there along with some additions. Although the primary screen is now a menu-driven control centre, anyone familiar with III+ can easily operate the program with little or no training.

For the beginner, dBASE IV includes a forty-five minute tutorial disk which users can refer to until they become proficient in building databases from the control centre. For the serious programmer, it provides over 440 commands and functions. Do they all work? This I cannot tell you, but those that I tried, did work.

Another great feature is its compatibility with dBASE III+. It can run any program written in III+. Also, programmers can move a library of programs over to dBASE IV with the assurance that massive changes will not have to be made.

The most serious shortfall that I found was that it is impossible to compile any IV program into an executable file.

(Continued on Page 2)

WAR CANCELS JANUARY MEETING

The NRC building on Sussex Drive will not be available to us for our January Annual General Meeting (AGM) due to a tightening of security brought about by the war in the Middle East. We were unable to find, on such short notice, an alternative facility. Therefore, there will be no January meeting. The location of future meetings will be announced in next month's Newsletter, on The PUB (the OPCUG's BBS) and on the Group's voice telephone line (723-1329).

Because the AGM will be held in February, the annual reports of the Executive, the financial report of the Group and all election-related information will appear in February's Newsletter.

* * *

FROM THE EDITOR

I would like to extend a big WELCOME HOME to David Smith, a dedicated long-time member of the OPCUG. David, who was working for the Candian Government in Baghdad, Iraq, returned safely to Ottawa this week. Welcome home David!

CONTENTS

.LANIMRET SIHT EDISNI DEPPART M'I !PLEH

DOES dBASE IV REALLY WORK?

(Continued from Page 1)

Ashton-Tate is currently developing a compiler that is supposed to accomplish this task, but no date has been set yet for its release. This creates an obstacle because, unlike the past, one cannot use Clipper or any other database to create an executable file.

How much disk space does it require? Here's the good news! Although IV is larger than III+, users can get by with 3.6MB of disk space if they are familiar with the product. A beginner will need 5.4MB to be able to consult the tutorial when necessary. A professional developer would use the developer's package which takes 8.4 MB of disk space.

A new feature of IV is its dynamic memory management system (DMMS) which reduces RAM usage to a minimum of approximately 384K. Other memory features, such as "DBHEAP", which allocates memory for overlays and applications, and an integrated disk cache option, which works with extended and expanded memory, improve its performance.

Other features include Query-by-Example (QBE) and a more flexible browse mode which allows the user to add data to a record or add a record without having to move back to the input screen. Access to an organized menu while in browse or edit mode provides the ability to organize data or create new indexes. QBE enables the user to view more fields and vary the width of columns.

The program offers multi-user capabilities, an eight-level security password, automatic record locking, auto-detection (to detect changes by other users), a user ID indicator, buffer protection against power failure and data encryption.

dBASE IV 1.1 was a latecomer to the market and now Ashton-Tate is trying aggressively to recapture lost ground. For example, if you own a registered copy of dBASE IV 1.0, Ashton-Tate will send you 1.1 at no extra cost. For any other registered dBASE users, the Company has set up a graduating price scale to enable them to buy dBASE IV at a lower-than-retail price. Ashton-Tate also wants to know of any problems that users may be experiencing with 1.1. Its Ottawa office telephone number is 234-3129.

Should you buy dBASE IV? The answer lies in your application. If your current software satisfies your needs and IV adds little improvement over it, you should weigh these factors against the $438 US to $499 US price of the standard edition. If you are a professional developer, Ashton-Tate has a developer's registry which you can join. However, spending about $789 US for a product that does not yet have a compiler could be money ill spent.

The bottom line is that dBASE IV 1.1 works! Ashton-Tate is trying hard to win back consumer confidence. My advice regarding this product (or on any other product) is to try it out before buying. Don't trust rumours, and don't always blame the manufacturer if you have problems.

dBASE IV 1.1 is a cradle-to-grave product. Ashton-Tate designed it to cover the widest possible market segment. A beginner can start and grow with it to become an advanced programmer. The product's pervasiveness and stability have given Ashton-Tate's dBASE line its current market success. Sticking to this formula may make it successful again.

* * *

VIRII FROM ABROAD

By Paul Cooper

The computer industry in Britain is being swarmed by virii coming from Eastern Europe, particularly from Bulgaria and Russia.

Part of the problem stems from the fact that Bulgaria does not believe in copyright and patent laws to protect software programmers. The virii are in themselves a form of protest against the authorities.

The Soviet Union has no copyright law and some software writers are already using them to punish people who steal programs. One such virus displays the message "Lovechild: in reward for stealing programs" on the screen.

The police in Bulgaria are aware of the problem but can do little to stop virus programmers since the country has no law against computer crime.

A British expert in the field recently came back from Bulgaria with 100 virii unknown to the West. These virii can do much damage by corrupting files and data stored on hard disks.

Less than twenty years ago, there were only 20 or so virus programs around, now there are hundreds. In Bulgaria, a new virus appears once a week.

The important thing for us to realize is that these virii can reach Canada within just a few days.

(Adapted from an article published in "The Independent", London, England.)

* * *

WP5.1 DATE FORMAT BUG

By Micheline Johnson

One of the more recent releases of WordPerfect 5.1 (8-20-90) contains a bug relating to the date format setup (Shift F1, 4, 2). No matter what format you set your date to and save it to the WP{WP}.SET file, the next time you run WordPerfect and display the date (Shift F5, 1), the default date format appears.

In the WP{WP}.SET file generated by the INSTALL and WP programs, releases dated 6-29-90 and 8-20-90 respectively, the date format is at a different offset from the one generated by the INSTALL and WP programs dated 4-17-90.

This date format bug, which also appeared in the January 1990 release, can be corrected by deleting the WP{WP}.SET file and creating a new one. To do this, re-install the printer (Shift F7, S) and re-enter all the setup parameters (Shift F1) such as Date Format, Location of Files, etc.

However, WP loses the date format again if you specify Lang: UK (Shift F8, 4, 4) in the Initial Codes (Shift F1, 4, 5) . To overcome this problem, do not choose UK. Instead, fool WP into thinking it is using the default US Speller and Thesaurus by renaming WP{WP}UK.LEX and .THS to WP{WP}US.LEX and .THS respectively (after transferring the US originals to a safe directory).

* * *

PROLOG: THE SHEER PLEASURE AND POWER OF IT

By José Campione

INTRODUCTION

Throughout the years, I have accumulated many subroutines in Turbo Pascal. When I decided to switch to Turbo Prolog (for the sheer pleasure and power...) I wanted to profit from them. Version 1.0 of the Owner's Handbook of Turbo Prolog indicated a succinct procedure for calling subroutines written in C. It also stated that "The process with Pascal (...) is similar". This statement no longer exists in the Reference Guide of PDC Prolog 3.2.

Was version 1.0 more powerful than 3.2? Well, not quite. First, notice that it said "Pascal", not "Turbo Pascal". Second, the PDC Owner's Handbook also indicates that "PDC Prolog allows the interface (...) with other languages - as long as those languages produce standard object modules (.OBJ files)". So, the problem in calling Pascal from Prolog is not in Prolog, it is in Turbo Pascal's inability to output OBJ code...

For some very well kept reason, Borland has decided to comment out the subroutine allowing the Turbo Pascal compiler to output files in OBJ format. They must have figured that compiling units into slick TPU files was all that Turbo Pascal required. Which is true, as long as you do not attempt the heresy of placing Turbo Pascal code at the service of another language...

I was about to give up in my efforts when a friend showed up with a copy of Peter Immarco's article published in the late Turbo Technix (Language Connections column, March/April 1988). There, Peter explains the use of TPU2OBJ.EXE which he describes as "a new Borland utility". My exhilaration started to fade when I could not find such utility in the original diskettes of the Turbo Pascal compilers 4.0 and 5.5. Then I decided to phone Borland. To my surprise, the technical expert on Turbo Pascal denied Borland's authorship, but he was kind enough to point that the utility I was mentioning was probably buried somewhere in Compuserve. I mentioned this in a message to John Whelan, and Mike Merritt had the extreme kindness of downloading TPU2OBJ from Compuserve and uploading it to the PUB. "Problem solved" as John said. And here is where this article starts, as a tribute to the resourcefulness of the PUB and its users.

This article pretends to be a light introduction to PDC Prolog programming, perhaps just enough to entice the reader to pursue further, and by himself, "the sheer pleasure (and power)" of thinking Prolog. For this reason, a succinct bibliography is suggested at the end. The strong procedural bias in this presentation is due to the background of both the author and the intended readers.

A BRIEF HISTORY OF PROLOG

Prolog (PROgramming in LOGic) was developed in the early 70's by Alain Colmerauer and his colleagues in Marseille. They were largely implementing abstract ideas on logic programming developed by Robert Kowalski in Edinburgh. Colmerauer and Kowalski seem to have worked together during one summer. Colmerauer also had a Canadian connection: he published in Montréal an article on "les systèmes Q" (systèmes québécois, peut-être...?) (Colmerauer, 1973).

Here is how David Warren from the University of Manchester describes his first experience with Prolog in 1974:
"Magic! (...) Deduction was being harnessed before my very eyes to produce effective computation. Declarative programming was truly programming on a higher plane! I had dimly seen the advantages in theory. Now Prolog had made them vividly real in practice. Never had I experienced such an ease in getting a complex program coded and running." (Sterling and Shapiro, 1986).

This element of "easiness" and "elegance" is pervasive to all aspects of Prolog programming. By the way, David Warren is the inventor of "tail-recursion optimization".

Many implementations of Prolog have been produced (Wisdom Prolog, Waterloo Prolog, Prolog-II, IC-Prolog, MU-Prolog, Arity Prolog, Micro-Prolog, etc. There is even a public domain PD Prolog (not to be confused with the PDC Prolog which superseded Turbo Prolog). "Core Prolog" or "Standard Prolog" refers to Prolog specifications as described in the book by Clocksin and Mellish (1984). This is also referred to as the "Edinburgh implementation".

This article is mainly concerned with PDC Prolog 3.2 which superseded Turbo Prolog versions 1.0, 1.1 and 2.0. Turbo Products never had the reputation of closely adhering to external standards, and Turbo Prolog (and PDC Prolog) are no exception. Because of the lack of metalinguistic abilities, some went even as far as to say that Turbo Prolog was not a Prolog at all... (But this is hardly an insult compared to the assertion believed by many that Prolog is some kind of LISP dialect...). Most implementations for mainframe machines correspond to interpreters instead of compilers and required massive memory capacity.

Borland did for Prolog what Henry Ford did for the car. 80,000 copies of Turbo Prolog were sold (which is not much compared to the numbers of copies sold for Turbo Pascal), but Prolog failed to become a "developmental" accepted program. This was undoubtedly the reason why the utilitarian wizards at Borland decided to discontinue their support to Turbo Prolog.

Thus, Turbo Prolog was returned to the Copenhagen gremlins that had been behind it since the very beginning, which revealed that Turbo Prolog, as Prolog itself, was not, had never been and had just failed to become a product suitable to the American dream... But how could it have succeeded, being primarily a French language, a language for the sheer pleasure of it...

A PROCEDURAL PERSPECTIVE PROLOG IN A NUTSHELL

In a heretic procedural nutshell, Prolog is a collection of calls to subroutines (called predicates) which can either fail or succeed. When a failing condition is met, the "failure" is passed upwards to the calling routine. If during this "back-tracking", a predicate is met which offers the possibility of an alternative solution, then the process is re-started "downward" from that point. The predicates usually describe facts or relations (rules) between objects.

Prolog's "intelligence" is its ability to infer new relations from more elementary ones. Some of Prolog predicates have the potential for more than one solution. If a call succeeds, the program proceeds to the next one; if it fails the program goes back to the nearest previous call generating alternative solutions. There it generates a new solution and tries again the subsequent predicates.

Predicates with the potential for several alternate solutions are said to establish backtrack points, and the ability of Prolog

(Continued on Page 4)

PROLOG

(Continued from Page 3)

to go back when a predicate fails is called "backtracking". It has two predicates (commands?) for controlling backtracking ("fail" and "the cut"). New user-defined predicates can be created by combining the primitive ones to build new ones called clauses. These user-created predicates can again be combined to create new ones and so on.

No variables can be directly passed between predicates (which also means that there is no such a thing as a common variable). Instead, Prolog implements a stylish way to store any kind of information during program execution: the dynamic database. This is just a fast way of storing, retrieving and erasing information from memory. All this could be presented from a different angle. Some experts would argue that in fact Prolog "is" a database of relations between objects.

(To be continued next month)

* * *

FILE BACKUP WITH PKZIP

By Gord Philips

PKZIP is a simple, yet effective, tool for backing up your important files. Using PKZIP's ability to select individual files, subdirectories or entire directories, you will be able to compress all of your important text, data, batch and configuration files into one dated ZIP file that may fit onto a single floppy backup diskette. (It is not really necessary to back up programs if you have kept the installation diskettes and/or the program's ZIP files that you have downloaded from the PUB.)

Keep a rotating series of backup diskettes. The oldest diskette is reused for the next backup. If you have some spare space on your hard disk, you may want to keep a copy of the latest ZIP backup on your hard disk in a directory such as C:\BACKUP. This way, you can quickly recover an individual dataset without having to fiddle with floppy diskettes.

Multiple backup diskettes are kept in case an earlier version of a file needs to be recovered. You determine how many backup versions (i.e. the number of backup floppy diskettes) that you need for your own purposes.

Date stamp each ZIP backup file with a dataset name such as 91JAN04.ZIP. This naming convention allows you to keep more than one ZIP backup file in your hard disk backup directory and/or on each floppy diskette (if there is enough room).

The following BAT file is a convenient way to do all this. It backs up all the files in the root directory of the C: drive and then selects the files and recurses the directories that are specified in the BKUP.LST ASCII-text dataset.

An example of BKUP.LST follows the BKUP.BAT listing. Instead of using the BKUP.LST file, you may prefer to change BKUP.BAT to specify the files and directories to be backed up as parameters on the PKZIP command line -- the PKZIP manual shows how to do this.

If your ZIP backup file is too large to fit onto a single floppy diskette, add another PKZIP command to the BKUP.BAT file and split the files and directories evenly between the two. This will be slightly less convenient because you will then require multiple floppy diskettes for each backup.

Backups will only be made if it is convenient. If the BKUP.BAT does not quite meet your needs, modify it so that it does. If you require something more sophisticated, try JOHNBACK (available on the PUB) or one of the commercial packages such as FASTBACK.

One more thing, in the event of a major hard disk failure, make sure that you have a copy of PKUNZIP on a floppy diskette so that you will be able to unZIP from your backup diskettes. If there is enough room, you may even want to copy PKUNZIP onto each backup diskette as a separate executable file (i.e. which is not part of the ZIP file).

TO BACKUP YOUR HD FILES

Use your text editor to create in ASCII the BKUP.LST file containing the names of files and directories to backup. Execute BKUP.BAT with today's date. For example, to create backup file 91JAN05.ZIP, enter:

```
BKUP 91jan05
```

TO RECOVER A SINGLE FILE

To recover C:\TEXT\FILE01.TXT from your ZIP backup file that was created on January 5, 1991, change to the directory where you want to recover the file and enter:

```
pkunzip a:\91jan05.zip text\file01.txt
```

TO RECOVER A DIRECTORY

To recover all the files and subdirectories in C:\TEXT from your ZIP backup file that was created on January 5, 1991, go to the directory where you want to do the recovery and enter:

```
pkunzip -D a:\91jan05.zip text\*.*
```

(Note: the -D tells PKZIP to create the necessary subdirectories. If -D is omitted, all files will be unZIPped into the current directory.)

BKUP.BAT

```
@ECHO OFF
IF NOT "%1"=="" GOTO CONTINUE
echo.
echo BKUP: Backup all hard disk data
echo files listed in
echo C:\UTIL\BKUP.LST to
echo C:\BACKUP and a floppy diskette.
echo.
echo Usage: BKUP yymmmddd
echo.
echo yymmmdd = the date of the backup
echo (e.g. 91JAN05)
echo.
GOTO EXIT
:CONTINUE
:Backup the root directory (without
:recursion).
:(Note: keep the -AP in upper case!)
PKZIP -AP C:\BACKUP\%1.ZIP C:\*.*
:Backup all the selected files and
:directories (with recursion).
:If you need to split your backup into
:multiple zip files, add more PKZIP
:commands here.
:(Note: keep -APR in upper case!)
PKZIP -APR C:\BACKUP\%1.ZIP
@C:\UTIL\BKUP.LST
echo.
echo PUT ZIP BACKUP DISKETTE IN
echo DRIVE A:
PAUSE
DEL A:*.ZIP
COPY C:\BACKUP\%1.ZIP A:
:EXIT
```

Here is an an example of a BKUP.LST.

```
c:\bat\*.*
c:\procomm\*.cmd
c:\procomm\*.prm
c:\progams\*.c
c:\wp51\*.set
c:\text\*.*
c:\letters\*.*
```

* * *

BACK TO BASICS

By Harry Gross

SEQUENTIAL FILES

In the last article on strings, we discussed the parts of a file specification. To review it quickly, the four parts are:

Drive - Drv$="C:"
Path - Pth$="\tutor\demo\"
Name - Nam$="example"
Extension - Ext$=".001"

There are three basic file types: sequential (or serial), random, and binary. For this article, we will discuss the sequential file.

As implied by its name, a squential file must be written to or read from in sequence. You start at the beginning and read through to the end; no jumping to the last chapter to see who the killer was.

In a sequential file, fields are separated by a comma, which can be seen when the file is examined with most text editors. On the other hand, records are separated by the ASCII characters 13 (CR or Carriage Return) and 10 (LF or Line Feed). These characters are not explicitly shown as is the comma, but the effect of the pair shows up as the next record, starting on a new line.

To start with, the file must be opened and given a file number. There are a few variations on the statement, but I will only illustrate one here. Once you get the hang of it, look in your manual for others.

```
OPEN Spec$ FOR OUTPUT AS #1
```

This tells the program the file specification to be used, how it is to be used, (write to a sequential file) and the file number (#1) to use.

Consider the following example, where we write three fields to one record.

```
A$="James": B$="Bond": C%=7
Q$=Q$ 'the quote symbol "
OPEN "A:test.dat" FOR OUTPUT AS #1
WRITE #1, A$,B$,C%                          ' (1)
PRINT #1, A$,B$,C%                          ' (2)
PRINT #1, A$ B$ C%                          ' (3)
PRINT #1, A$ "," B$ "," C%                  ' (4)
PRINT #1, Q$+A$+Q$ "," Q$+B$+Q$ "," Q$ C% Q$   ' (5)
CLOSE #1
```

Each of the output statements will produce the following format on disc:

"James","Bond",7

(1) Strings are in quotes, numbers are not. The fields are delimited by commas.

James Bond 7

(2) All output is printed with tabs on 14 wide column zones, but nothing between the values to indicate different fields.

JamesBond 7

(3) The output is packed. The space before the seven is reserved for the sign which is omitted if positive, included if negative. There are no field delimiters.

James,Bond, 7

(4) The output is now packed, with commas between the fields. However there are no quotes around strings. This may or may not be an advantage depending on the application.

"James","Bond"," 7"

(5) The output is now inside quotes and comma delimited. Note that C%, having started life as an integer, has now been converted to a string and must be so treated when read back in. Reading it back in as a numerical variable will produce a value of 0.

The last statement closes the file. No further writing operations may be performed on it and the file number is released for other uses.

Before a sequential file opened for writing may be read, it must be closed and re-opened for input. Reading and writing to a file in the same set of operations is not possible. Only one or the other is allowed.

What do you do if a string has a comma in it? Consider the string

S$="What's up, Doc?"

If we use PRINT #1,S$, we get

What's up, Doc?

in the file. However, on reading it in (to be discussed later), we would get

S$="What's up"

with the comma vanishing and "Doc?" becoming another variable. To preserve the integrity of the variable, we should use:

```
PRINT #1,Q$+S$+Q$  or
WRITE #1,S$
```

to give us

"What's up, Doc?"

Similar treatment can be afforded strings that may have quotation marks inside them by breaking them up and putting them together again. However, there is a small mess produced in writing them, and a larger mess possible in reading them. So skip the use of quotes for now.

Any ASCII character in the range decimal 33 to 255 may be used in a serial file, but avoid the quote (") and observe the problems with the comma, ASCII 44.

For short files, we don't really have to worry about the structure of the file. We can put material in any old way as long as, in general, we read it with the same variable type as it was printed to the file.

In our next article, we will discuss in detail the structure of sequential files.

* * *

BEGINNERS' CORNER

By Eric Clyde

PROCOMM SET-UP FOR THE PUB

The following sequence of commands, including a script file, should help those of you who would like to use Procomm to log on to the OPCUG's Bulletin Board, The PUB. Procomm is available in the Group's software library.

Start the program by typing Procomm at the DOS prompt. Various parameters must then be checked in the Setup Screen (Alt-F4).

When the setup menu appears, go into Modem Setup. The default dialing code (2) is ATDT. If you have a pulse dial line, change this to ATDP. These letter must be capitalized.

General Setup

You can eliminate sound effects and exploding windows. Any changes you make must be saved to disk (S).

Line Settings (Alt-P)

Some of the settings depend on your hardware, others on the Bulletin Board being accessed. I use 2400,N,8,1,COM3. The 2400 signifies a 2400 baud modem, and the COM3 is the communications port I use. For The PUB, you must use N,8,1.

The Dialing Directory

Press <Alt D> to access the directory. It will already contain about ten preset numbers. You may either overwrite one of the existing numbers with The PUB's, or use a blank line. In both cases, type 'R' to revise an entry. You will be asked for the number of the entry to be revised. Fill in the blanks as you are prompted for them. For example:

Name: Ottawa PC Users' Group
Number: 565-1880
Baud: 2400 (that of your modem)
Parity: N
Databits: 8
Stop Bits: 1
Echo On? N
Command File: Leave this blank unless you wish to automate the sign-on and password steps.
Save Entry to Disk? (Y/N): Y

Dialing the PUB

After completing the above steps, type <Alt D> to access the dialing directory and n (where 'n' is the entry number for the BBS you wish to dial).

Assuming a line is free, you should be logged onto The PUB and will be asked for your full name and password (if you have been on before). If you have a color monitor and The PUB only sends part of the screen (from the left margin to the end of each text line) in colour, go into Setup <Alt S>, choose General options, then option O - ANSI Compatibility, and change it to 3.x by hitting the spaceba. Escape and save your configuration.

Downloading Files from the PUB

Downloading files in Procomm is simple. After typing in the name of the file you wish to download, press the <Pg Dn> key to choose a protocol. If you download text files, use the ASCII protocol. For executable files, use X-Modem, Y-Modem, or Y-Modem Batch. I normally use Y-Modem Batch because The PUB automatically types in the filename for me.

If you have problems, call me (749-2387).

* * *

COMPUTER TRANSLATIONS

By Jean Fortier

We'll take a look this month at various types of printers and the vocabulary associated with them.

continuous paper	- papier en continu
daisy-wheel printer	- imprimante à marguerite
dot matrix printer	- imprimante par points
draft mode	- impression rapide
font	- police (de caractères)
impact printer	- imprimante à impact
laser printer	- imprimante à laser
line printer	- imprimante ligne par ligne
letter quality	- qualité lettre
parallel printer	- imprimante parallèle
pin wheel	- roue à picots
printer driver	- logiciel de gestion d'imprimante
ribbon	- ruban
serial printer	- imprimante série
type, to:	- frapper, taper (des caractères)
typeface	- oeil (d'un caractère)

In the next article, we'll look at pointing devices, such as mice, trackballs, mouse pens and light pens.

* * *

THE LAN TUTORIAL SERIES

By Aaron Brenner

PART 1: Buying a LAN

This short article kicks off LAN Magazine's new series of "clip-and-save" tutorials about LANs. Each month we will print an easy-to-read tutorial -- aimed at users new to networking -- covering one aspect of LAN purchase, installation and management.

This first tutorial is a very basic introduction to the issues involved in buying a LAN. Along the way is an overview of the components of a LAN and a list of the next 12 topics to be covered.

The four parts to come should provide you with short, easy-to- understand introductory pamphlet about the principles of local area networking.

A Definition

A LAN is a data communications network spanning a limited geographical area, a few miles at most. It allows users to share information and computer resources, including mass data storage, backup facilities, software, printers, plotters and processors.

Typically, a LAN is made up of network interface cards (circuit boards) that fit inside the connected computers, cable to connect these computers, protocol software to move data from computer to computer, user interface software to connect user and network, and operating system software to actually service users' needs for things like files and printers.

Why Buy a LAN?

LANs require a certain mind set, something different from traditional MIS (Management of Information Services) thinking. Once a LAN is installed, things like initiative, democracy, participation, communication and independence take over. Hierarchy, dependence, regulation and isolation are thrown out the window.

If you have the right mind set, the four best reasons to buy a LAN are:

1) Communication - A LAN connects the people in your company. Every form of discourse is possible, from electronic "yellow sticky things" to formal legal briefs. People like to communicate.

(Continued on Page 7)

THE LAN TUTORIAL SERIES PART 1

(Continued from Page 6)

2) Democracy - A LAN distributes your company's computer resources to everyone connected. Once the LAN is installed, everyone from the mail clerk to the chief executive officer will want, and should have, access to it.

3) Productivity - A LAN's ability to share computer resources and information easily helps people do their jobs quickly, efficiently and with less hassle. The LAN will quickly become the heart of your business. It's the heart of ours.

4) Savings - A LAN saves money by allowing users to share expensive computer resources -- printers, plotters, hard disks, WORM drives, CPUs, software, etc.

If you don't have the right mind set, the four best reasons not to buy a LAN are:

1) Communication - Connecting all the people in your company might let them talk to each other. Who knows, they might plot your overthrow.

2) Democracy - Distributing resources will give everyone in the company a measure of power. Who knows, they might not do what you tell them.

3) Productivity - Doing the job in new and better ways might lead to the elimination of dull, tedious work. Who knows, it might mean the permanent elimination of your job.

4) Savings - Sharing expensive computer resources saves you money. Who knows, you might have to save money all the time.

LAN Components

Buying and installing a LAN is not simple. There are many things to think about.

The following are twelve aspects that cover the basics of LAN purchase and installation. Study them carefully before committing yourself or your company to such an undertaking.

1. The OSI Model, which stands for the Open Systems Interconnection model of the International Standards Organization, is a useful categorization of the different parts of a LAN. It is an overview of how a network works.
2. The Access Method is the way the network arbitrates which device may use the cable and for how long. It is necessary since two devices can't talk at the same time. Different access methods provide different network performance and reliability.
3. The Interface Card is the device that connects the computer to the cable. These vary by type, size, speed and much more. Performance is a key issue.
4. The Cabling is the physical connection between networked devices. Fiber optic, coaxial and twisted-pair are the main choices. Each has advantages and drawbacks.
5. LAN Protocols are software that run in the computer and on the network interface card. They provide the means for shipping data between devices. Certain sets of protocols are good for certain applications. Which you choose depends upon what you use your network for.
6. The LAN Operating System is the software that resides in the computer. It provides the interface between the user or application and the network. The key issues here are performance, compatibility and ease of use for everyone involved.
7. The File Server stores and distributes program and data files to be shared by users on the network. It is a hardware/software combination heavily dependent on the LAN operating system and the type of work you are doing.
8. Network Printing allows many users to share one or more printing devices. Some LAN operating systems do it better than others. Sometimes you will require special network printing utilities.
9. Tape backup, done regularly, maintains data integrity on a LAN by recording data on tape instead of disk. Key issues include capacity, speed, compatibility and ease of use.
10. LAN Security covers the methods used to protect data from corruption by unknowing users, accidents and intruders. These include physical security, encryption and passwords. But the type of security you use depends mostly on the type of work you are doing on your network.
11. Bridges and gateways connect networks. Each uses different methods with different results. Bridges connect networks at a lower level than gateways, making them more versatile. On the other hand, gateways connect networks that bridges can't. Performance and compatiblity are the key issues.
12. LAN Management is the name given to the best job in the world: taking care of the network. Different LANs provide different levels of management to make the job easier. Your level of skill and confidence will be crucial here.

Application Specific

The type of LAN you buy depends primarily on the work it will do. Before evaluating different vendor options, assess your company's computer needs and resources, present and future. As much as possible, conserve your present computing power, even if you plan to upgrade.

Will the network be used mostly to share peripherals like printers and hard disks? If so, access methods and performance are less important than reliability and ease of use.

Will the network be used mostly for large database access? If so, performance is paramount.

Will the network be used mostly for communications and electronic mail? If so, wide-range standards compatibility may be the most important issue in your decision.

Usually, you want the network to do everything: start out peripheral sharing, add databases then connect to mainframes. Thus, growth potential and standards are very important to your decision, since you're laying a foundation upon which you will build.

Other overall considerations include: the education of users (beginners and experts); the types of computers you are connecting (PCs, minicomputers and/or mainframes); and the amount of money you have (lots or a little.)

Unfortunately, no LAN does everything. Vendors make compromises, sacrificing ease of use for performance, performance for compatibility or vice versa. Since this is the case, get to know exactly what you want before you buy.

* * *

OTTAWA PC USERS' GROUP

EXECUTIVE

Chairman	David Terroux	238-4895
Past Chairman	Stu Moxley	832-1278 or 832-4378
Treasurer	Tony Frith	671-0401
Secretary	Claude Jarry	521-3366
Newsletter Editor	Bonnie Carter	236-1015
Software Librarian	Chris Taylor	723-1329
BBS System Operator	Jean Fortier	236-1015
Convener	Douglas Poulter	745-8768
Membership Chairman	Carl-Henri Gomez	731-1462
Hardware/Software Broker	Terry Mahoney	225-2630 226-2615 FAX

ASSISTANTS

Assistant Newsletter Editor	Jean Fortier	236-1015
Newsletter Assistants	Chris Taylor	723-1329
	Ann Falkner	722-3226
	David Smith	837-9291
Software Assistant	John Ings	235-8132
Membership Assistant	Marc Riou	733-2092
BEGINNERS' SESSIONS	Eric Clyde	749-2387

DISK OF THE MONTH

One-year subscription (10 Disks) 5.25"	$25.00
One-year subscription (10 Disks) 3.5"	$35.00
Individual 3.5" disk surcharge	$2.00

THE OTTAWA PC USERS' GROUP MEMBERSHIP APPLICATION - Please Print

Last Name: _____ First Name: _____

Mailing Address: _____

Postal Code: _____ Telephone - Home: _____ Office: _____

Profession: _____ Date of Birth: _____ Sex: M _____ F _____

Disk of the Month: YES _____ NO _____ Size: 5 1/4" _____ 3 1/2" _____ Amount Enclosed $ _____

Are you: A new member? _____ Renewing your membership? _____ Sponsor's Name: _____

What in particular interests you in the Group? _____

Can you help in Group activities? Check off the activities that apply: Programming Language Instruction _____

Newsletter Input _____ Memberships _____ Software Library _____ Promotion/Publicity _____

Hardware Techniques _____ Meeting Locations _____ Agendas & Speakers _____ Advertising _____

Bulletin Board _____ Other _____

What hardware/software do you own and/or use? _____