



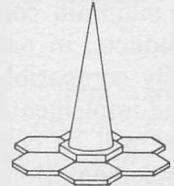
I. P. Sharp

# newsletter

## SHARP APL/PC Ready for Delivery in February

I. P. Sharp Associates has announced an important addition to its products for the Global Information Centre: SHARP APL for the IBM Personal Computer, to be available February 1984. SHARP APL/PC is a full implementation of the SHARP APL available for mainframes. Because the two versions are absolutely compatible, applications can be developed in either the mainframe or the PC environment, and can migrate easily to the other.

SHARP APL/PC includes processors for: accessing both SHARP APL and DOS files, communicating asynchronously with mainframes—including any SHARP APL system, driving a plotter, using non-APL programs, and managing full-screen input/output.



SHARP APL/PC runs under IBM PC DOS, version 1.1 or 2.0, and requires 384K of memory.

SHARP APL/PC costs US\$375, with sizeable discounts for quantity orders. The basic package includes:

- SHARP APL/PC system on diskette
- Utility programs on diskette
- *SHARP APL/PC Handbook*, for both the beginning user and the application programmer
- APL stickers for the PC's keybuttons
- SHARP APL/PC keyboard template for quick reference
- *SHARP APL Pocket Reference*

You may also wish to order:

- APL character ROM, which enables an IBM PC to display APL characters, priced at US\$25
- *SHARP APL Reference Manual*, priced at US\$18

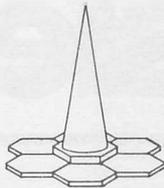
Orders for February delivery are now being accepted by all I. P. Sharp offices. If you are a customer of the I. P. Sharp timesharing service, you can bill the cost of SHARP APL/PC to your timesharing account. In Canada, the United States, the United Kingdom or Australia, you can charge your order to your VISA, MasterCard, or American Express account.

For more information about SHARP APL/PC, ask for the fact sheet from the Micro-computer Products Group at I. P. Sharp's international headquarters, or from your I. P. Sharp representative. ■

### in this issue

SHARP APL/PC Ready for Delivery in February .....	1
SHARP APL for the IBM Personal Computer .....	2
The Stockholm Airport Information System Threat Analysis for Oil Companies and Utilities .....	5
Using MABRA for Conference Administration	6
New Local Dial-up Number.....	7
Keep Informed with <i>World Report</i> and <i>Aurora Report</i> .....	8
Far East Data Bases.....	8
SUPERPLOT: Cheaper and Easier than Before	10
Predicting the Future with SIFT.....	11
Boston, Coventry, Düsseldorf, Frankfurt, Melbourne, Zurich.....	13
Bookends .....	14
<b>Technical Supplement 47</b>	
Using S-tasks to Monitor Systems.....	T1
Making Things Fit, Part 3.....	T3

## SHARP APL for the IBM Personal Computer



SHARP APL/PC is the latest product I. P. Sharp offers for the Global Information Centre. I. P. Sharp strives to maintain compatibility across its range of products; in particular, SHARP APL/PC is fully compatible with SHARP APL/370 (the implementation for System/370 mainframes) and will be compatible with future implementations.

SHARP APL/PC establishes in your PC an *operating environment* in which you use the SHARP APL *programming language* to process data and to command the PC.

From the SHARP APL/PC operating environment, you can enter SHARP APL statements or execute SHARP APL programs to:

- Process data
- Store and retrieve data and programs on floppy or hard disks
- Store and retrieve data created or used by other applications—APL or not
- Control and use your PC's keyboard and display screen
- Communicate with a timesharing service, another PC, a serial plotter, a printer, or another external device

### SHARP APL on the PC

SHARP APL/PC offers *exactly the same language* as SHARP APL/370. Although some system functions, system variables, and system commands have been modified or redefined for the PC, all APL constructs available in one environment are available in the other: all primitive functions and operators; and all data types, including packages and enclosed arrays. As well, two important segments of the system interface—event-trapping and files—remain the same. What this means is that programs written in one environment can be moved to the other easily. It also means that SHARP APL data is not affected by the PC's 16-bit architecture; for example, a data object's size is limited only by the amount of memory available.

This exact compatibility has been achieved by basing SHARP APL/PC on a **System/370 emulator**. Rather than rewriting SHARP APL for the PC, I. P. Sharp's microcomputer products development team wrote an emulator, which makes SHARP APL look like a "native" program to the PC, and makes the PC look like a System/370 to SHARP APL. Thus, SHARP APL/PC incorporates a thoroughly tested implementation—SHARP APL/370—nearly in its entirety.

An emulator also ensures that SHARP APL/PC is fully compatible with future SHARP APL products. For example, you will be able to move SHARP APL/PC software easily to the SHARP APL system for the IBM PC XT/370, when it is available.

### Minimum configuration

To use SHARP APL on your IBM Personal Computer, you need this minimum configuration:

- IBM PC or PC XT (If you have a PC XT, you can have it field-upgraded to an XT/370.)
- 384K of memory (The operating system and SHARP APL take up about 300K in total; so whatever is left is available for the active workspace and session log. For example, if your PC has 512K of memory, your active workspace and session log share about 200K.)
- 5¼-inch diskette drive
- IBM monochrome display/parallel printer adapter; or the IBM colour/graphics adapter; and a monitor that functions with either adapter
- APL character ROM (installed on the display adapter and sold separately by I.P. Sharp. This ROM is not necessary if you have the APL\*PLUS/PC custom ROM sold by STSC Inc.)
- IBM Personal Computer Disk Operating System (DOS), version 1.1 or 2.0

### SHARP APL/PC software

An operating SHARP APL/PC system includes: the interpreter, the file system, the session manager, several auxiliary processors, and utility software.

### ... SHARP APL/PC

The **SHARP APL/PC file system** gives you access to SHARP APL files on your PC's diskettes. These files are compatible with mainframe files. You can also store data and programs in a **saved workspace**—a copy of the active workspace on diskette—or in a **native file**—a DOS-compatible file created outside the SHARP APL/PC environment.

The **SHARP APL/PC session manager** controls input from the keyboard and output to the screen, records input and output transactions on its session log, and controls the log itself. You use your Personal Computer's DELETE or BACKSPACE key to delete characters from anywhere in an input line, the INSERT key to insert characters anywhere, and the RETURN key to enter the line, that is, to send it to APL. You can grab lines from the session log, edit them, and reenter them.

You specify to the session manager how certain keybutton combinations are to be interpreted, and how certain characters are to be displayed. You can also program as many as 128 keybutton combinations to be function keys.

You can specify to the session manager what portions of the session log you want to see, and what portions are or are not to be erased.

SHARP APL/PC includes several auxiliary processors. An **auxiliary processor (AP)** is a program that runs in the PC beside SHARP APL, and that SHARP APL communicates with through the **shared variable processor (SVP)**. You can use the APs supplied with SHARP APL/PC, or you can write an AP to do anything the PC can do.

The basic SHARP APL/PC product includes these APs:

- asynchronous communications AP
- native file AP
- parallel printer AP
- function call AP
- full screen AP

The **asynchronous communications auxiliary processor (ACOM)** handles real-time communications with external devices through the PC's RS232 ports. ACOM uses the worldwide standard asynchronous (start/stop) protocol.

Thus, a SHARP APL/PC program can communicate with, for example, another PC, a serially connected printer or plotter, or a time-sharing service (including the I. P. Sharp time-sharing service or any other SHARP APL system).

The **native file auxiliary processor (DOSF)** handles DOS files for SHARP APL/PC programs. SHARP APL programs use DOSF to store data in, or retrieve data from, files created by application programs outside SHARP APL. Thus a SHARP APL program can share data with, for example, 1-2-3 or VisiCalc.

The **parallel printer auxiliary processor (PPRT)** handles a printer attached to the PC's parallel printer port. Thus, a SHARP APL/PC program can drive a parallel printer directly and print APL characters.

The **function call auxiliary processor (CALL)** passes data to a non-APL program, executes it, and returns data from it.

The **full-screen auxiliary processor (FSCR)** handles the PC's monochrome or colour display separately from the session manager. A SHARP APL/PC application program can use FSCR to set up specialized input and output forms—such as a sales slip, an invoice, or an inventory record.

### Utility software

SHARP APL/PC includes utility programs written and tested by I. P. Sharp. These programs are generalized building blocks to be used or adapted by application programmers for developing application systems. ■

## The Stockholm Airport Information System

Arlanda, Stockholm's international airport, has installed a new airport information system. InterSystems BV, the Dutch subsidiary of I. P. Sharp Associates, has supplied this integrated data collection and distribution utility under contract to the Swedish Board of Civil Aviation.

Now passengers and the general public can view up-to-the-minute flight information on video display monitors or other display equipment. As well, airline and airport staff—concerned with services such as aircraft maintenance, refuelling, baggage handling, in-flight catering, customs, and immigration—have use of sophisticated online information collection and display facilities.

The system consists of a dual central computer configuration (connected, in turn, to airline and other computer systems for automatic exchange of data), and a dual MEGATEX video display and distribution system. MEGATEX is a proprietary product of InterSystems. (See box.)

MEGATEX at Arlanda International Airport maintains several hundred pages of real-time information, and supports more than 500 display units located throughout the airport.

The system, using the latest computer and communications technology, features:

- High performance and reliability
- Ease of maintenance
- Facilities for future expansion
- Continuous 24-hour-a-day operation
- Low level of human intervention

The system was installed in September at Arlanda's new domestic terminal, and will be in service at the existing international terminal in February 1984.

InterSystems has supplied similar systems to Schiphol International Airport in Amsterdam and to Kastrup International Airport in Copenhagen.

For further information on the Integrated Airport Information System or the MEGATEX Video Display Information System, contact the Airport Systems Group of InterSystems, or your I. P. Sharp representative. ■

### MEGATEX

MEGATEX is a flexible, general-purpose video distribution system with wide application wherever video displays are connected to computer systems, for example, in:

- Airports
- Stock exchanges
- Brokerage systems
- Dealing rooms

MEGATEX maintains a data base of information organized into formatted pages. One or more host computers connected to MEGATEX may update the information pages in real time. MEGATEX distributes the information pages to locally or remotely connected video display monitors. These monitors may display pre-selected pages, or they may have page selection devices, such as push-button numeric keypads. Once a page is selected, the information on a MEGATEX monitor is maintained in real time; that is, each time the MEGATEX data base is updated by a host computer, information updates are automatically transmitted to all video display monitors displaying that page.

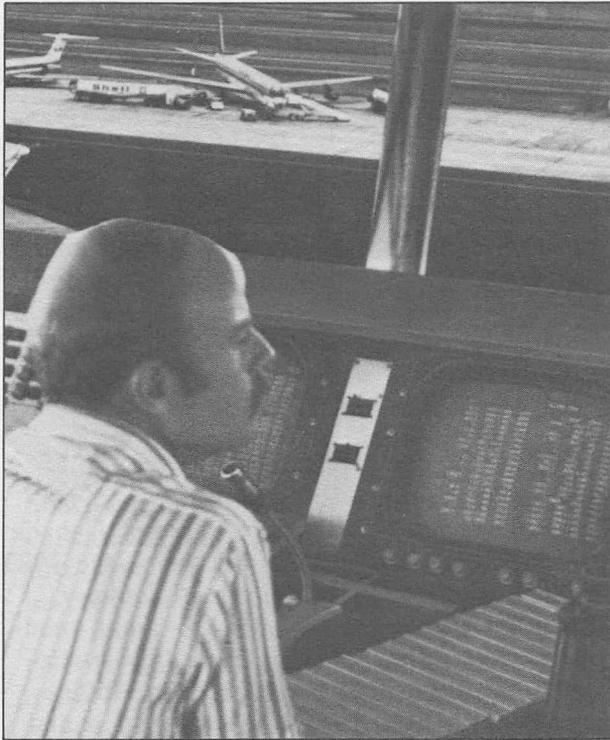
A MEGATEX configuration consists of a central controller containing a video data base and a network of video controllers, each supporting up to 16 video displays. The central controller is connected to one or more host computers.

MEGATEX is a modular system available in configurations supporting from eight to 2 000 monitors. Each system may maintain up to 10 000 information pages. Transmission speeds of up to 30 000 characters per second are available. Colour and graphics facilities are planned.

MEGATEX is proven technology and available now. For further information, contact InterSystems in Amsterdam, or your I. P. Sharp representative.

*Laurie Howard, Amsterdam*

## ... Stockholm Airport Information System



*MEGATEX rapidly and reliably disseminates large amounts of information to thousands of people at airports. Service information for ground support and airline personnel is displayed on video monitors in control towers, airline offices, maintenance areas and other outlying areas.*

## Threat Analysis for Oil Companies and Utilities

For about four years, I. P. Sharp Associates has been offering programs for threat analysis, sometimes known as risk analysis. This form of investigation and reporting examines "incidents" which might occur, and consequently reduce the value of the assets of an organization.

I. P. Sharp's risk analysis methodology is based on a model written in SHARP APL. This model is defined afresh for each new application; only its most basic building blocks are permanent. This means that the user defines assets, the perils to which the assets

are subject, the ways in which incidents of increasing degrees of severity cause loss to the organization, appropriate countermeasures, and the various ways each of these might reduce the level of vulnerability.

In the near future, threat analysis methodology will be applied to nuclear power stations and off-shore structures, primarily for oil companies and utilities. I. P. Sharp will offer front-end processors, specific to each application. These software modules will use the particular information available within each of these well-defined environments, and convert it to the type of data needed for the threat analysis package. Using only data which is readily accessible, the analysis may be automatically carried out as far as the introduction of proposed countermeasures. Once the variables associated with a countermeasure are entered, it will be evaluated in the usual way and the cost/benefit ratio calculated.

### Threat analysis procedure

The first part of threat analysis, which involves several steps, measures the vulnerability of the assets or their **exposure**. This is expressed in terms of the statistically computed expected loss per year to be experienced by the owners of the assets. Once it is determined, the analysis proceeds to calculate the amount by which the exposure may be reduced by the introduction of various countermeasures. As the result of a countermeasure, the incidents which cause the exposure may either be prevented altogether, or else be forced to occur less severely. The total reduction in exposure which would result from using a particular countermeasure is taken to be the benefit accruing to it. Each proposed countermeasure is evaluated by comparing its benefits to its costs using standard cost/benefit analysis techniques.

The first step in threat analysis is to determine assets. For off-shore structures, the assets may be categorized in terms of their function. Initially three categories of assets are envisaged: seaworthiness (flotation), purpose (drilling, collecting oil, etc.), and the physical structure itself (housing). If any of the three is not working to capacity, the structure will be limited in its use or will stop functioning completely. The major peril to off-shore struc-

### ... Threat Analysis

tures is ship collisions. Another is ruptures to the flotation devices. Determining how these and other perils affect the assets, and to what degree, is the purpose of performing the threat analysis.

For nuclear power stations, the exposure to some incidents cannot be measured adequately in dollars, the usual unit. For example, the threat may be evaluated in millirems (the unit of radiation). I. P. Sharp's threat analysis package can measure the exposures in these two units simultaneously.

### Flexible design

I. P. Sharp's threat analysis methodology has proven to be totally flexible in design, and applicable to many different situations. Originally developed to protect assets associated with computer-based information systems, the same techniques have been used successfully to determine protection for bridges from ship collisions. By performing an initial, very detailed analysis of the available information concerning the bridge structure, geometry, and traffic (both underneath and over the bridge), the data required by the threat analysis model can be produced automatically. In this case, and in situations involving nuclear power stations and off-shore structures, the user is never asked a single question about assets, threats, and events. This is in direct contrast to the usual practice in performing threat analysis, with or without computer assistance.

When SHARP APL for the IBM Personal Computer is introduced in February, a total threat analysis package will be available on the PC. This package may include one or more appropriate, specific, front-end processors. At that time, if you have highly sensitive data, or are concerned with the confidentiality of results, you will be able to run the system in your own protected environment.

Because the techniques involved in threat analysis are so flexible and powerful, and are used to explore future, uncertain events, there is a wide range of possible uses. Other applications being considered include social and economic development plans, especially for lesser developed countries.

If you are interested in any of these applications of threat analysis or have a unique application, contact David Bonyun, at I. P. Sharp's Ottawa office, or your I. P. Sharp representative. ■

*David Bonyun, Ottawa*

---

## customer application

---

### Using MABRA for Conference Administration



The Progressive Conservative Party of Canada held two large conventions this year—the 1983 General Meeting, in Winnipeg, in January, with 5000 registrants, and the National Leadership Convention, in Ottawa, in June, with 8000 registrants. I. P. Sharp's Ottawa office developed the system for the registration and administration of these political conferences. From these quite complex applications evolved a thorough and flexible package which may be quickly and easily customized for your convention or conference.

The conference system is based on MABRA, I. P. Sharp's data management system. Since MABRA is flexible, simple to use, and very easy to learn, the registration staff at Progressive Conservative Headquarters in Ottawa realized its advantages, and opted for the system. Within hours of determining its specifications, the system was set up, ready for data entry, and the staff fully trained.

MABRA's capability to store the data in real fields, together with calculations based on this data in pseudofields, formed a flexible data base design providing many useful tools for information handling.

The systems for the two conventions differed only in minor details, since the rules and structures of both were very similar, as would be the case for most conventions.

A mandatory requirement for the General Meeting in Winnipeg was the availability of a communications network. All planning and pre-registration was done in the Progressive Conservative Headquarters in Ottawa, and the data was accessed in Winnipeg during the

### ... Using MABRA

convention via the I. P. Sharp Network.

The data base contained many types of information about each registrant: name; mailing address; represented province, electoral district or club; voting status; constitutional category; hotel accommodations; sessions attended; and financial information such as amounts paid, due, and outstanding.

Financial information was calculated at a somewhat detailed level, since the fees were determined by a multitiered structure based on the status of the individual registrant. Different fees were levied for senior delegates and alternates, youth delegates and alternates, observers, spouses of delegates, and other spouses.

### MABRA responds quickly

In addition to MABRA's fast and powerful report-generating capability, the registration staff were impressed with MABRA's ability to respond quickly to queries about any aspect of the stored information. For example, while still organizing the conventions, they could answer, in minutes, questions such as: "How many delegates or alternates from the electoral district of Ottawa-Carleton have pre-registered?" "How many delegates from any given district have had their status certified by the credentials committee, but have not yet registered?" "How many registrants have fees outstanding?" And the most frequently asked question was: "How many people have registered to date?"

Then the staff could choose the amount of information they wished to report, based on each query. The detail of response varied from simply reporting the number found, to doing detailed crosstabulations and generating formatted reports broken down by any number of fields for each person. The staff could also specify any combination of sort fields in these reports; that is, by district, or by surname within district within province by postal code.

Data entry was accomplished through a full-screen interface, designed for HDS 108 terminals. Screens were designed to resemble the completed registration forms, which were mailed to Ottawa by all delegates, alternates, and other groups of registrants in a pre-registration process. During the convention,

data entry became a real-time application where volunteer staff verified the arrival of each registrant.

### Custom reports

Using report generators and customized reports in MABRA, many reports were designed, including periodic confirmation of account reports which were sent to all registrants, mailing labels for the mailing of information, listings formatted as address labels of registered delegates to date for candidates or speakers at the convention, name badges for all groups of registrants, voter lists for polling stations, registration check-in lists, and accounting documents and reconciliation reports of cash received and accounts outstanding.

The registration system was used right up to voting time. Then a team of professional auditors used the registration system's pre-audited voter lists and randomly generated voter numbers to manage the voting process manually.

The automated system was used after the conference to "close the accounting books" on the attendees. The system was also a source of statistics on attendance which will be useful to the Progressive Conservative Party for some time to come.

For more information on how MABRA can help you manage your collection of records, whether for a conference or another application, contact your I. P. Sharp representative. ■

*Ronna Button, Ottawa*

---

**communication  
news**

---

### New Local Dial-up Number

**Helsinki**

120 cps 021 452 1133

## Keep Informed with *World Report* and *Aurora Report*

Two international news bulletins are now available on the I. P. Sharp timesharing service in public workspaces and through INFOMAGIC, I. P. Sharp's conversational data base retrieval service. *World Report* and *Aurora Report* are published by Immedia Telematics Inc. of Hudson, Quebec. *World Report* covers international news as it breaks, and *Aurora Report* covers developments in the international computer and communications industries. There is no subscription fee for access to either news bulletin.

### *World Report*

*World Report* is edited by a team of seasoned journalists with international news gathering experience in newsmagazines, newspapers, wire services, radio, and television.

Direct reporting by Immedia News Network staff is supplemented by the international broadcasting services of the British Broadcasting Corporation, Radio France International, Radio Canada International, and Radio Australia.

More timely than the daily newspaper and available any time, *World Report* keeps business and government professionals informed throughout the working day. *World Report* is updated five times daily: at 00:00, 10:00, 15:00, 18:00, and 21:00 GMT. These regular editions are supplemented by special bulletins on urgent, breaking news.

You can access *World Report* through INFOMAGIC, or by typing:

```
)LOAD 985 WORLD
```

The headlines of the current edition appear first. You then select the stories that interest you by entering their headline numbers, separated by spaces. There are four to six major news reports in each edition as well as a section of news briefs which summarizes other events.

You may read any or all the news stories, in any sequence you desire. You can return to the headlines by entering *H*, get the masthead by entering *M*, or end the session by entering *S*.

### *Aurora Report*

*Aurora Report* is a weekly news bulletin, updated Wednesdays. It concentrates on the computer and communications industries.

You can access the *Aurora Report* through INFOMAGIC, or by typing:

```
)LOAD 985 AURORA
```

Then follow the same operating instructions as those for *World Report*. ■

## Far East Data Bases

I. P. Sharp has available two new data bases: Hong Kong Stock Exchange and Singapore Corporate Statistics. In addition, Singapore market rates have been added to the Money Market Rates data base. A brief synopsis of each of these data bases follows. For a more detailed description, consult the *Financial and Economic Newsletter*, October 1983, Volume 3, Number 4.

### Hong Kong Stock Exchange Data Base

Daily trading statistics and descriptive information are now available for over 100 stocks traded on the three major Hong Kong stock exchanges: Far East, Hong Kong, and Kam Ngan (gold and silver). The Hong Kong Stock Exchange (HKSTOCK) data base contains the following facts for each stock:

High	Highest price of all three exchanges
Low	Lowest price of all three exchanges
Close	Close price from the Far East exchange (the most volatile)
Volume	Aggregate of volume from the three exchanges

Descriptive information for the securities includes: security symbol, security number, description (full name), date security added, date of last quote, date of last trade, delist indicator, and industrial classification.

The data is available daily from January 1983. The data base is updated every day, at 21:00 Hong Kong time, with the most current



I. P. Sharp

# newsletter

## technical supplement 47

### Using S-tasks to Monitor Systems

This article describes a simple S-task application to ensure that update programs are run for three public data bases and that everything proceeds properly. Without the aid of S-tasks, this would involve signing on to each account to check, for example, if crash workspaces have been produced by the update and associated N-tasks.

Since signing on to these various accounts and checking them can be boring, time-consuming, and repetitive, it is an ideal task for a computer. But in APL, it is not possible to enter `)LIB` under program control to check for crash workspaces, or to enter `)XLOAD CONTINUE` to check for unexpected interruptions.

S-tasks solved the problem perfectly. Now, in about three minutes, the night's activities on four accounts can be checked, and any problems can then be investigated.

S-tasks are interfaces with APL, almost exactly like T-tasks, but accept input from Auxiliary Processor 1. AP1 is part of the SHARP APL operating system and itself accepts input from other tasks. Using a shared variable, AP1 passes that input onto APL, and APL's responses back to the S-task initiator. A detailed description of S-tasks can be found in *SATN-39* and the *SHARP APL Pocket Reference*.

The functions in this article were written to solve a specific problem. You might find the functions serve as a good introduction to S-tasks, and then branch out from these, according to your own requirements.

The following example is a subset of the operational program. To make the principles of writing an S-task function clear, the program

has been simplified. This version signs on to an account to check for N-task crash workspaces, and to check the *CONTINUE* workspace for any exceptional conditions.

In the following, two functions—*DAILY* and *READSTASKTRANS*—are used. *DAILY* sets up the S-task, signs on, gives input to, and prints output from the S-task. *READSTASKTRANS* reads the output from the S-task, and when the whole response has been received, delivers it back to the function *DAILY*.

#### *DAILY*

```

▽ DAILY;□IO;SHARE;H
[1] □IO←0
[2] ←1 □SVC 'SHARE'
[3] ←1 1 1 1 □SVC 'SHARE'
[4] H←□AV[0 0 0 0]
[5] SHARE+H,')',ACCOUNTPASS
[6] □+READSTASKTRANS
[7] SHARE+H,')PW+80'  aMAKE OUTPUT FIT ON A SCREEN
[8] SHARE+H,')LIB'
[9] □+READSTASKTRANS
[10] □+'CHECKING CONTINUE WS DATE AND □ER'
[11] SHARE+H,')XLOAD CONTINUE'
[12] □+READSTASKTRANS
[13] SHARE+H,')ER'
[14] □+READSTASKTRANS
[15] SHARE+H,')OFF'
[16] -READSTASKTRANS aFINISH JOB..DON'T PRINT
[17] a ACCOUNTING INFO
[18] -□SVR 'SHARE'
▽

```

The first instruction sets `□IO` to 0, which is convenient when dealing with S-task protocol, but not necessary (see *SATN-39*). Then comes the offer to share the variable *SHARE* with AP1. *SHARE* need not be previously defined. `1 1 1 1 □SVC 'SHARE'` enforces a reversing half-duplex communication. Thus neither partner can set or use the variable until the other has respectively used or set it. Neither partner misses the actions of the other, nor can either proceed until the other has completed its processing. No other control or interlock is necessary.

## ... Using S-tasks

Input to an S-task must be preceded by a four-element array (*H*), each element of which is equal to the zero (IDLE) character ( $\square$ AV[0]). If the third element of *H* is zero (IDLE), the input is passed to APL. The other characters must always be zero (IDLE) characters. *SHARE* is set to *H*, then catenated with the account number and password (e.g. 1234567:PASS). Note that to keep passwords in a program or workspace is usually regarded as a fairly serious security exposure.

The program *READSTASKTRANS* is then called, and returns the result of the signon. This application assumes that the signon was a success. After the sign-on message has been printed, the S-task is asked to *XLOAD CONTINUE* (to avoid activating the  $\square$ LX), the date stamp is printed to find when it was saved, and  $\square$ ER is printed. Then the task is signed off, and the shared variable offer retracted. It is important to sign off in an orderly fashion, so that no *CONTINUE* workspace is created by abnormal termination of the S-task. Note the use of  $\rightarrow$  which returns its left argument. Since there is no left argument, nothing gets printed, and the use of junk variables is avoided.

### READSTASKTRANS

```

V R+READSTASKTRANS;□IO;B;TEMP;L
[1]  □IO←0
[2]  R←''
[3]  C1:
[4]  TEMP←SHARE⍲      SHARE IS DEFINED IN DAILY
[5]  C3:
[6]  B←(8⍲2)⍲□AV⍲TEMP[5]
[7]  →(B[6]=1)/END ⍲PROMPT FOR INPUT →C B[6]
[8]  L←256.□AV.2+TEMP
[9]  R←R,2+L+TEMP ⍲DON'T WANT TO SEE PREFIX
[10] TEMP←L+TEMP
[11] →(C⍲TEMP)/C3
[12] →C1
[13] END: X

```

Let us begin with an explanation of why this function is necessary. When you are sitting at a terminal, APL returns your output one line at a time. However, if the S-task returned information to you in the same way, it would put an unnecessary burden on the APL system. Therefore, the S-task controller will

catenate several lines together. This necessitates informing the partner of what follows. Each line from an S-task starts with two bytes which tell how long the following segment is in bytes, including the length bytes. The next four bytes or characters are the line's prefix. For the purposes of this application, the only important one is the fourth one. Among other things, it indicates if the following segment is normal output or a prompt. *READSTASKTRANS* keeps reading the share until a prompt for input is encountered.

Let us now go through the function *READSTASKTRANS*.  $\square$ IO is again set to 0, and the result variable initialized. The global variable *SHARE* is used, and the S-task is free to set it again. In line 6, the sixth character of the segment header is resolved into bits. The program exits if the prefix signals a prompt for input. If not, the length of the line is calculated, and the line is catenated to the result after dropping the length and prefix information. The processed line is then dropped from *TEMP*, and the next line is analysed. If no more data is left in *TEMP*, the shared variable is used again to get the next segment. Otherwise, the next segment in *TEMP* is analysed.

These functions can easily be amended to meet your specific requirements. A few assumptions have been made: for example, it is assumed that another task on the same account has not already used the same clone ID; and that the sign-on was successful.

Following is an actual session using the previous programs, showing what the output looks like. Note that here the sign-on is to my own account. When you sign on to your own account for a second time (assuming that your account has multiple sign-on permission), it is not necessary to use a password. ■

Sandra Eadie, Toronto

## ... Using S-tasks

```
)LOAD STASK
SAVED 19.17.53 09/09/83
DAILY
OPR: 1 NEWS, ITEM 59 NOW IMPLEMENTED.
4690) 13.54.13 09/12/83 SANDRAEADIE
*** 2 TASKS ***

SHARP APL SERVICE

CONCEPT    CONTINUE    DDEF        STASK

CHECKING CONTINUE WS DATE AND ER
SAVED 13.53.29 09/12/83

2 SYNTAX ERROR
ARBIN
^
```

---

## Making Things Fit, Part 3

In the first two parts of this series, we considered ways to fit non-rectangular data in rectangular arrays, and large amounts of data in finite workspaces. In this article, we will suggest ways to fit data within the limitations of files.

### Making things fit in files

SHARP APL files are very flexible. So flexible, in fact, that you may be wondering what limitations must be overcome. There are three minor limitations and two major ones.

Of the minor file limitations, one applies to all accounts. SHARP APL limits the number of files that an individual task may have tied at the same time. At most installations, the limit is 20 ties. To get around this limit, you can untie files when you are finished using them.

The two other minor limitations may vary from one account to another. Each account has a file quota limit which limits the number of files an account can own, and a file reservation limit which limits the total number of bytes that can be reserved. Both numbers are set when an account is created, and should be set high enough so that problems are unlikely. Each can be changed by the operations staff. You can trap *FILE QUOTA USED UP* and *FILE*

*RESERVATION ERROR*, and then send a message to the operator to increase the appropriate limit.

What are the major limitations on the use of files? Unlike workspaces, there is no fixed upper limit on the size of file components or on the entire file. But there are *practical* limits on both. A component can be as big as a clear workspace. However, if you want your programs to execute quickly, a much smaller size is desirable. A file can be as big as the disk space allotted to the file system. But since disk drives cost money, and most installations charge users for disk storage, keeping a file as small as possible is desirable.

As mentioned in the previous article, one of the chief ways of segmenting data to make it fit into a finite workspace is to put it on file. Here, too, the data must be split into parts. Since the SHARP APL file system provides a great deal of latitude, a designer is faced with two questions:

- 1) How small should a file component be?
- 2) How big should a file component be?

Due to the way the file system works, an important question to consider first is:

- 3) How can file components be kept at their minimum size?

Each of these questions is addressed in this article.

### How can file components be kept at their minimum size?

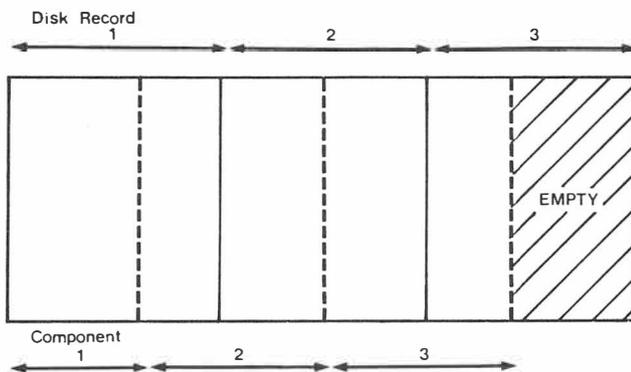
This question obviously implies that components can take up more space than they need to. While the other two questions in this article deal with the strategy of segmentation, this one deals with that of volume alteration.

The previous article suggested ways to minimize the volume of variables and functions stored in the active workspace, and apply to all variables and functions stored in a file as well. There is one additional volume distortion problem that applies only to files, however—the infamous growing REPLACE. Also the benefits of the scaling technique are more pronounced when applied to data in a file. To scale data is to move the decimal point.

## ... Making Things Fit

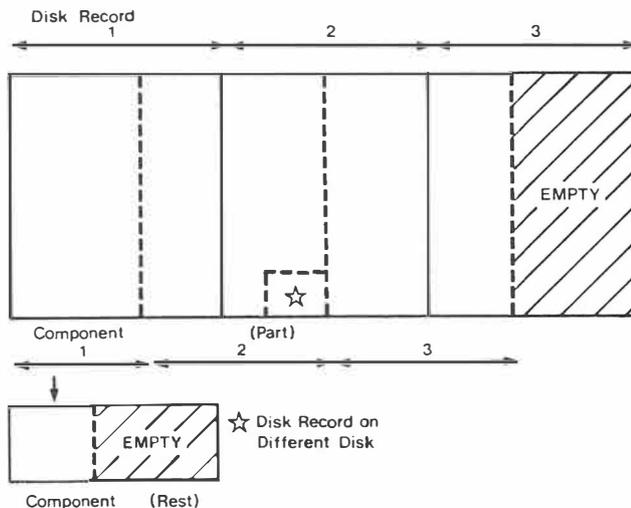
We need to recall how components are stored in a file. When they are first appended, consecutive components are stored next to one another within a disk record. Depending on where the previous component ends and the size of the component, it may span one or more disk records.

Consider the case of a file containing four 500-element integer vectors. The file would resemble the following:



### COMPONENTS SPANNING DISK RECORDS

What if we want to replace the second component with a vector of 501 integers? Obviously, it won't fit. The original space is used, less a few bytes for a pointer. This pointer indicates the new disk record on a different disk where the remainder of the component is stored. The empty space in this extra disk record is then available for further growth of *that component only*. The next diagram shows the revised file after the growing `REPLACE`.



### GROWING REPLACE

You can imagine how much space is wasted if this process occurs hundreds or thousands of times. Not only is space wasted, but execution time as well. If every component in a file has been changed just once by a growing `REPLACE`, twice as many disk records must be fetched to process the entire file.

How can you avoid growing `REPLACE`'s? First, build up arrays in the workspace, and write them to file when done. The following functions demonstrate the wrong and right approaches to this problem.

```

▽ BUILDΔWRONG;LINE
[1] TEXT←''
[2] LOOP:LINE+,□ ◇ +(0=ρLINE)ρEND
[3] ((□READ TIE,COMP),LINE,CR) □REPLACE TIE,COMP
[4] →LOOP
[5] END:▽

```

```

▽ BUILDΔRIGHT;TEXT;LINE
[1] TEXT←''
[2] LOOP:LINE+,□ ◇ +(0=ρLINE)ρEND
[3] TEXT←TEXT,LINE,CR
[4] →LOOP
[5] END:TEXT □REPLACE TIE,COMP ▽

```

What if you know a file has become bloated due to growing `REPLACE`'s? You can wait for the operations staff to run the file reorganization utility, which is typically run each week-end. This compresses all files back into the minimum space required to store them. If the file is consuming a lot of space, or will receive heavy usage, you may want to reorganize it yourself immediately. You can do this by simply copying the file, deleting the original, and renaming the copy. The next function shows you how.

```

▽ FREORG OLD;CTR;LMT;NEW;NAME
[1] NEW←1+⌈ /0,□NUMS
[2] 'FREORGTEMP' □CREATE NEW
[3] (1+2+□SIZE OLD) □RESIZE NEW
[4] CTR←0 ◇ LMT←-1+1+1+□SIZE OLD
[5] LOOP:→(LMT<CTR+CTR+1)ρEND
[6] (□READ OLD,CTR) □APPEND NEW
[7] →LOOP
[8] END:NAME←,□NAMES[□NUMS;OLD;]
[9] NAME □ERASE OLD
[10] NAME □RENAME NEW
[11] □UNTIE NEW ◇ NAME □TIE OLD ▽

```

Sometimes it really is necessary to allow file components to grow. Is there a way to minimize the effects of growing `REPLACE`'s? You can pre-allocate space for components. The

## ... Making Things Fit

next function ensures that components have room to grow *without* requiring growing `□REPLACE`'s. It initializes the components to non-empty arrays, and then replaces them with empty arrays. The storage used by the first array is left unused, when it is replaced by the empty array. Components which are smaller than the one originally appended can replace the empty array without causing a growing `□REPLACE`.

```
▽ TIE←SIZE FBUILD NAME;DATA;CTR;LMT
[1] TIE←1+⌈/0,□NUMS ◇ NAME □CREATE TIE
[2] DATA←\SIZE[1] ◇ LMT←SIZE[2] ◇ CTR←0
[3] LOOP:→(LMT<CTR+CTR+1)ρEND
[4] DATA □APPEND TIE
[5] (10) □REPLACE TIE,CTR
[6] →LOOP
[7] END: ▽
```

Note that this extra pre-allocated storage is available only until the file reorganization utility is run (typically each weekend). This means that this technique is most useful when a file is being built using growing `□REPLACE` during a single session.

While growing `□REPLACE`'s are a special problem of files, the concept of scaling applies both to storage in files and in the active workspace. The most typical scaling operation involves moving the decimal point two positions to the right so that currency (often expressed as dollars and cents) is expressed in whole units. Then the numbers can be demoted from floating-point form to fixed-point form. Since the storage required is cut in half, this produces substantial space savings when large quantities of numbers are stored in a file. This technique can also save space in a workspace, but the results are less dramatic.

You might think that this is a case of trading execution time for storage space, but this need not be so. If all of the currency quantities you are working with are less than 21 474 836.47, your numbers will never be converted to floating-point form. So, you won't incur conversion costs, and you can do all your arithmetic in fixed-point form, which will actually save execution time. Moreover, the scaling feature of `□FMT` makes it possible to display scaled data without actually converting

it back to its original form. The following expression shows how it works.

```
, 'K-2 C F14.2' □FMT 1234567 987654321 12
12,345.67 9,876,543.21 0.12
```

## How small should a file component be?

The SHARP APL file system allows components to differ in rank, shape, data type, and size. It does a great job of mapping such components into fixed-length records stored on disk. While the way it is done is optimal for most users, the file system can use up unnecessary storage or execution time for an unwary few. The SHARP APL file system places as many APL components in a disk record as possible. (Some APL file systems put each component in a separate record on disk, even if only a small part of the record is actually needed to store the variable.)

How big is a disk record? On the I. P. Sharp timesharing service, it is currently 3156 bytes. At any SHARP APL site, you can determine the current disk record size by:

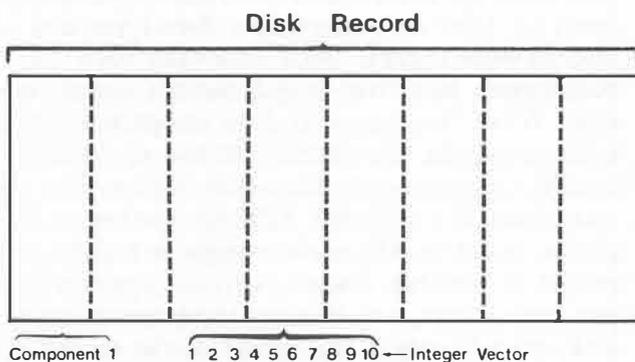
```
'TEMPORARY' □CREATE 9
(10) □APPEND 9
RECSIZE←.5×1+2+□SIZE 9
```

The first time you append a component to a newly created file, two disk records are written. One contains the data, the other stores a directory to the components. Thus the initial file size must be divided by two to find the disk record size.

How many arrays can you get in one disk record? Consider the case of a 10-element integer vector. Such a vector takes 56 bytes in the workspace. Dividing 56 into 3156 gives 56.35, but that is not the answer, because only 44 components of this type are actually stored in one disk record. Each component has audit trail data and overhead stored with it, which takes approximately 14 additional bytes. The directory record takes 4 bytes. And there are also eight bytes of header information at the beginning of each disk record, leaving 3148 bytes for components. So, the *SHARP APL File System* manual (p. 33) says:

## ... Making Things Fit

On average, the number of bytes of file space required to hold an APL value that occupies  $N$  bytes of workspace storage is  $1.04 \times (20 + N)$ . File storage amounts are always multiples of the physical record size (3156 bytes) used by the file system.



### DISK RECORDS AND FILE COMPONENTS

Remember these two important points when estimating the optimum size of a file component:

- 1) Variables need more storage when stored in a file than in a workspace.
- 2) The exact amount of space required depends on the particular array, and whatever else is stored in the disk record. The formula from the manual gives a reasonable approximation.

Since several APL components may be stored in a disk record, it is possible that the file system will read one or more components for free. The file system always fetches an entire disk record and stores it in one of its own memory areas. If a disk record fetched for you is still in memory when you request another component stored in that same record, no disk operation will be done. Instead, the data will simply be moved from the file system's memory area to your active workspace. This takes less execution time.

When the file system has a relatively large amount of working space available in proportion to the number of users signed on, it is possible to read some components for free. When the amount of space allocated to the file system is relatively small, or the number of users signed on is relatively large, the space

your disk record was using will normally be given to another account between your requests to read a file. Thus the file system cannot read any component for free.

In fact, when user load is moderate to high, small components are a definite disadvantage. Consider the case of 44 10-element integer vectors, all stored in a single disk record. Under a light user load, the disk record may be fetched when the first vector is read, and subsequent vectors are read from memory. But under heavy user load, the disk record may be fetched *each time* one of the vectors is read. That means transferring up to  $44 \times 3156$  bytes from disk, even though only 3156 bytes are actually used. In contrast, if the 44 vectors had been reshaped into a single matrix, at most 3156 bytes would be transferred, since only a single disk operation would be required.

When estimating the optimum size of a file component, important points to consider are:

- 1) Under a light user load, sequential processing of a file proceeds at optimum speed.
- 2) Under a moderate to heavy user load, reading lots of small components takes more execution time than reading the same amount of data stored in big components.

Thus we find the answer to our original question posed at the beginning of this section. To obtain optimum processing speed, the average file component should be (at least) *as big as two disk records*. ■

*Robert Metzger, Rochester*

*Editor's note:* This article on "Making Things Fit in Files" will be continued in *Technical Supplement 48*.

### ... Far East Data Bases

information available at the close of the exchanges that day. There is no subscription fee to access this data.

Computact (H.K.) Ltd. supplies the stock data as well as data on the Hang Seng index together with Keith and Company, a member of the Far East and Hong Kong exchanges. They supply the same information to local stockbrokers on diskette.

You can access the HKSTOCK data base through both 39 *MAGIC* and 55 *RETRIEVE*. In 39 *MAGIC*, type *HKSTOCK 'DESCRIBE'* to obtain a general description, or type *HKSTOCK 'DIRECTORY'* to obtain a complete directory for this data base.

### Singapore Corporate Statistics Data Base

The SINGCORP data base contains corporate financial statistics for over 250 companies listed on the Stock Exchange of Singapore. The National University of Singapore supplies approximately 250 annual facts for each company, from 1973, in the categories of:

- General particulars
- Profit and loss
- Operating expenses
- Extraordinary items
- Shareholders' funds
- Reserves
- Long-term liabilities
- Current liabilities
- Fixed assets
- Current assets
- Investments and advances
- Intangibles and other assets
- Financial ratios
- Miscellaneous

For access to the SINGCORP data base, there is an initial subscription fee of US\$500, plus an annual fee of US\$500. You can access SINGCORP through 39 *MAGIC*, 55 *RETRIEVE*, or *SES MARKETSCAN*.

### Singapore Added to the Money Market Rates Data Base

The Money Market Rates (MRATE) data base now includes the following market rates for Singapore:

**Singapore Prime Lending Rate:** Rate at which major Singapore banks are willing to lend Singapore dollars to their best customers.

**Asian Dollar Offer and Bid Rates:** Rates at which banks in Singapore are prepared to lend (in the case of offer rates) or accept deposits (in the case of bid rates) in United States dollars. The rates are quoted for periods of seven days, and one, two, three, six, nine, and twelve months.

**Other Asian Currency Rates:** Interbank rates for Japanese yen, West German marks, and Swiss francs. These are middle rates, a midpoint of the offer and bid rates, and are quoted for periods of one, three, six, and twelve months.

The data is available daily, from 30 May 1983, and is provided by the *Business Times* of Singapore. The data base is updated each morning (between 01:00 and 04:00 Singapore time) with the rates for the previous day. There is no surcharge for use of the MRATE data base.

You can access the MRATE data base through 39 *MAGIC*. In addition to the market rates for Singapore, you have access to 246 daily and weekly money market rates for several countries, as well as Eurocurrency rates and interbank currency rates for various currencies. To obtain a complete list of codes, type *MRATE 'DIRECTORY'*. ■

## SUPERPLOT: Cheaper and Easier than Before

SUPERPLOT is I. P. Sharp's general-purpose business graphics system. Past newsletter articles have outlined SUPERPLOT's power, ease of use, flexibility, and friendliness. Now as a result of continuing efforts to lower the cost and increase the ease of using SUPERPLOT, I. P. Sharp has dramatically reduced the cost of the PLOT function in Version 5, and introduced a new system to produce simple, standard plots.

### Reduced costs

The cost of the average plot has been reduced by approximately \$10. You still pay for specifying the options you want, and for the characters transmitted to your plotter or terminal, but the N-task used by the PLOT command costs you nothing. This radical step brings SUPERPLOT into direct price competition with other commercial plotting packages—packages which cannot offer the same plotting power and easy access to hundreds of public data bases that I. P. Sharp provides. You can combine these public data bases, your own data collections, and results of computations and analyses, with a powerful and inexpensive graphics tool, all on the I. P. Sharp timesharing service.

For production plots (plots that are run repeatedly with different data, but with the same options), costs may be reduced even further by using packaged plotstates which you create and store. These packages contain all the SUPERPLOT option settings, but not the data. In the development of a complex plot, the cost of setting the plot options can be substantial. With packaged plotstates, you can save the plot options and redefine them at any time at a fraction of the original cost.

### SUPERQUICK

SUPERQUICK takes advantage of packaged plotstates to help you produce simple, inexpensive plots with a minimum of effort.

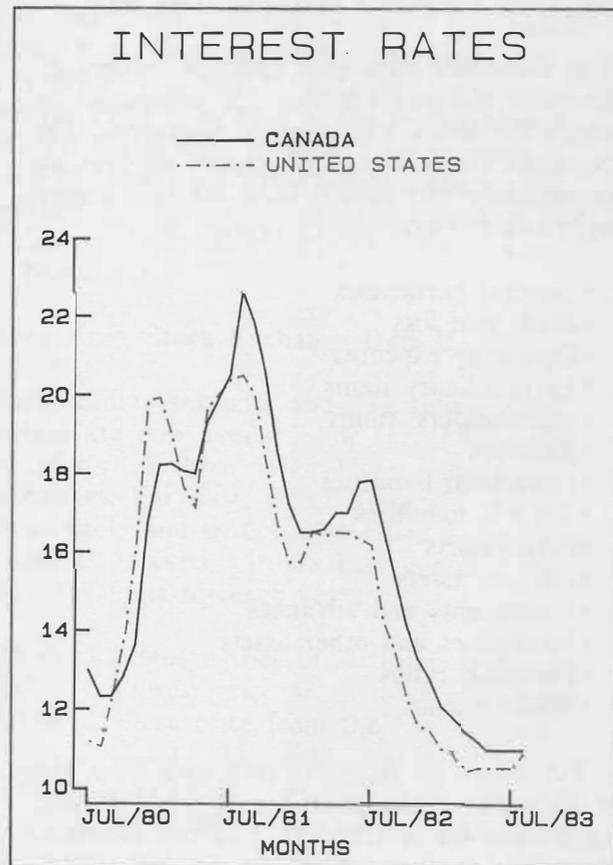
SUPERQUICK provides standard option settings appropriate to your plotting device for most popular graph types (line, point, bar, and pie). You can produce a plot by typing a single

command (for example, *PLOTPIE X1 AND X2*), and then answering a few questions. The basic SUPERPLOT options are set for you; you don't have to know anything about setting them. If you *are* familiar with SUPERPLOT, you may set additional options in the usual way within the SUPERQUICK sequence.

These and other recent SUPERPLOT developments are documented in the online SUPERPLOT NEWS facility. For more information, contact your I. P. Sharp representative or the MAILBOX group SUPQ.

The following plot was produced with SUPERQUICK, using data from the Money Market Rates data base. ■

Linda Furrow, Calgary



SOURCE: MRATE DATA BASE

## Predicting the Future with SIFT

*"If you can look into the seeds of time  
and say which grains will grow and which  
will not, speak then to me..." Macbeth*

The ability to predict the future no longer resides with the select few, as in Macbeth's time. Now you can chart the future course of time series using Sharp Interactive Forecasting Techniques (SIFT). SIFT brings together a wide range of time series analysis and forecasting techniques that enable you to forecast a series of interest to you. The series could be data from your own company (such as product sales), possibly stored using MAGICSTORE, I. P. Sharp's time series data storage and analysis package. Alternatively, the series could originate from I. P. Sharp's public data bases and be retrieved using MAGIC.

Time series analysis is based on the principle that future values of a series can be determined by examining its past values. In general, a series is considered to be composed of four components: randomness (or noise), trend, seasonality, and cycle. Your series may be composed of each of these components to a greater or lesser extent. In analysing the series you determine, as accurately as possible, the importance of each of these components for your particular series.

SIFT provides preliminary techniques that help you determine whether your series does indeed include trend, seasonality, or cycle (all series are considered to include randomness). You can then choose the appropriate analysis technique from: moving averages, exponential smoothing, curve fitting, decomposition, harmonic smoothing, trend-cycle analysis, and generalized adaptive filtering. If you don't know which technique to use, SIFT can recommend the ones appropriate for your data.

Once you decide on a method, you can arrive at a forecast very simply, using only a few basic commands. Or you can try various techniques, and have the results reported in tables, before deciding on the best forecast.

### Using SIFT

SIFT is available in the public workspace 39 SIFT or in 39 MAGIC. The variables DESCRIBE

and SIFTHOW in 39 SIFT give details on how to use SIFT. SIFT is completely interactive, prompting you for answers throughout. At every prompt, you can type HELP for information on how to answer the prompt and on how to proceed through the SIFT session. Also available on line are brief descriptions of the individual analysis techniques that can help you decide whether a technique is appropriate.

The following example shows how you can use SIFT to predict how certain grains will "grow" in the coming months. This sample session shows a basic sequence for obtaining a forecast. The data, retrieved from the International Financial Statistics (IFS) data base, is a series of Canadian wheat wholesale export prices (in U.S. dollars/bushel) from January 1978 to December 1982.

In answer to the first prompt, you specify the data timeframe. Then the WHEAT series is defined as the series to be analysed. If you use the FORECAST command, you are prompted for information necessary for calculating a set of forecasts (i.e. the forecast period and the analysis techniques). Here forecasts are produced for January to June 1983, using the one analysis, Holt-Winters' exponential smoothing (ESHW).

The default display in the following example includes a table of statistics that summarizes the results over the entire series, and the forecasts for the series.

```
)LOAD 39 SIFT
SAVED 19.14.46 08/25/83
)COPY MAGICDATA WHEAT
SAVED 15.12.30 10/20/83
SIFT
* TIME SERIES DEFINITION: MONTHLY,
  DATED 1 78 TO 12 82
* TIME SERIES COMMAND: DEFINE
* DEFINITION MODE? VARIABLE
* NAME OF VARIABLE: WHEAT
* TIME SERIES COMMAND: FORECAST
* FORECAST TO WHAT PERIOD? 6 83
* WHICH ANALYSES? ESHW
```

## ... Predicting the Future with SIFT

\* TIME SERIES COMMAND: STOP  
 ---EXITING SIFT---

---HOLT-WINTERS' LINEAR AND SEASONAL EXPONENTIAL SMOOTHING---

\* ENTER VALUES FOR ALPHA BETA GAMMA: .5 .2 .2  
 +-----+  
 +MEAN % ERROR (MPE) OR BIAS -0.27%/+  
 +MEAN SQUARED ERROR (MSE) 0.069405611 +  
 +MEAN ABSOLUTE % ERROR (MAPE) 4.26%/+  
 +-----+

	FORECAST
JAN/83	4.83
FEB/83	4.78
MAR/83	4.80
APR/83	4.74
MAY/83	4.68
JUN/83	4.72

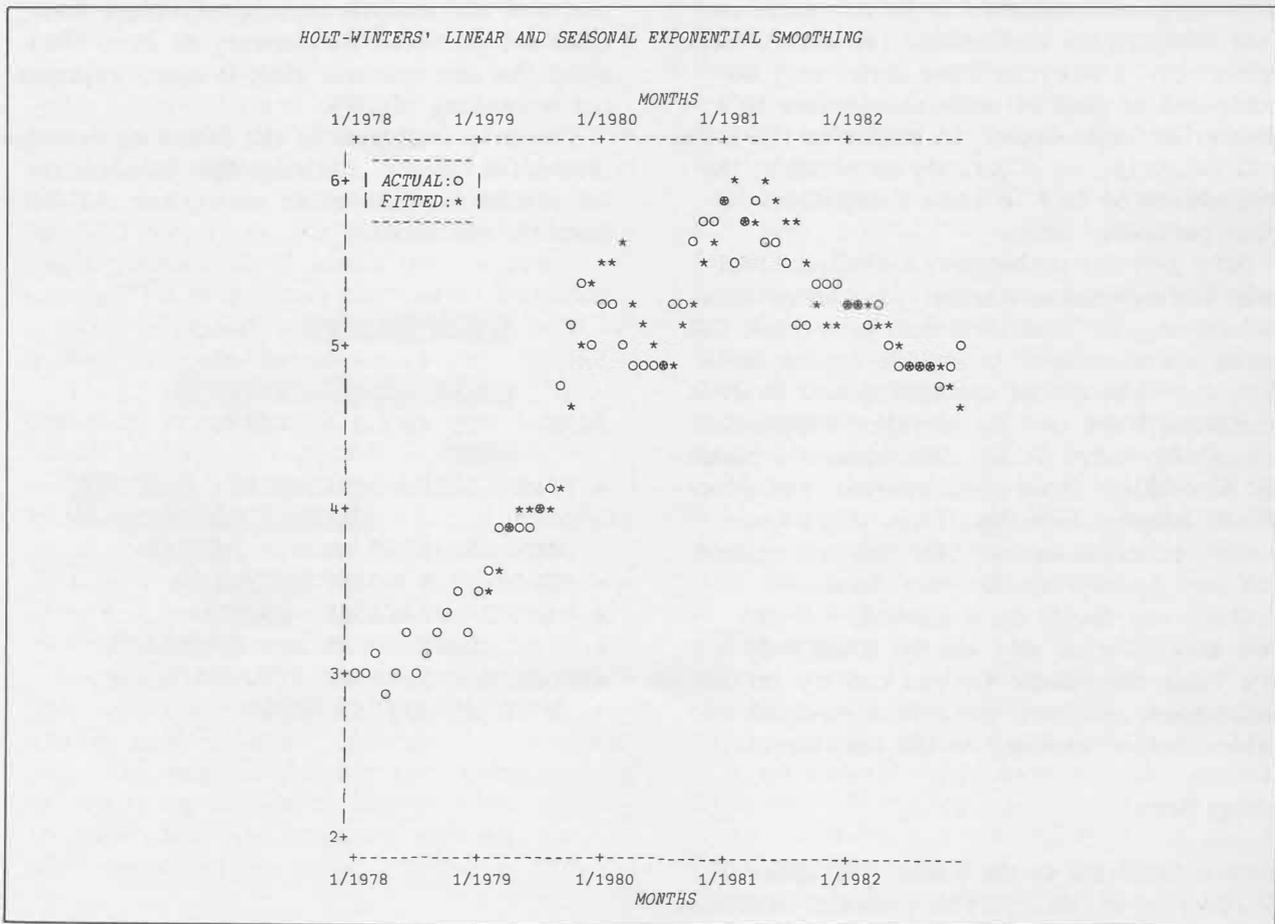
SIFT is fully documented in the *SIFT Users' Guide*, where both the analysis techniques and SIFT commands are described in more detail. To obtain a copy, or for any other questions you may have regarding SIFT or time series analysis, contact your I. P. Sharp representative or the MAILBOX group STATQ. ■

Caroline Haythornthwaite, Toronto

*"Stay, you imperfect speakers, tell me more."*

You can request additional output (for example, a plot of the fitted and actual values for the series) with the command *SHOW*.

\* TIME SERIES COMMAND: SHOW  
 \* WHICH OPTION(S)? PLOT



## Boston

**Larry Brennan** is the new branch manager in Boston. Before joining I. P. Sharp, he was marketing manager for Atex Inc. in Bedford, Massachusetts, a subsidiary of Eastman Kodak, which markets text processing and advertising systems to the publishing industry.

Larry studied at INSEAD, the European Institute of Business Administration, in Fontainebleau, France, and was a consultant to the Organisation for Economic Co-operation and Development in Paris.

A man of varied talents, Larry started his career as an urban planner and worked in Europe as a planner, consultant, and teacher. ■

## Coventry

**Michael Martinek** has recently joined I. P. Sharp as manager of the Coventry branch. He earned his B.Sc. in physics from the University of Durham and his Ph.D. in physics from the University of Newcastle.



Michael brings to his position a wealth of business experience, including operations research, production control and management, corporate planning, warehousing and distribution, and general management. He has worked for I.C.I. Paints Division, Ryman (office equipment), Chloride Gent (fire alarms, clocks, call systems), and Admiral Sportswear.

Before joining I. P. Sharp, Michael worked as an independent consultant with data processing experience in programming, analysis, system specification and hardware investment justification. ■

## Düsseldorf

**Sally Drew** has recently been appointed acting branch manager in Düsseldorf. After working for Air France in sales, she joined I. P. Sharp's London office in 1979, with responsibility for education in the United Kingdom and customer support for Kodak. In 1982, she worked in the Oslo office providing educational and general customer support. Then she moved on to the Paris office, primarily to develop and market the ICIS petrochemical data base.



Sally has been working in the Düsseldorf office since April of this year, where she has been responsible for supporting customers and marketing data bases. ■

## Frankfurt

**Gert Osterburg** is the manager of our new office in Frankfurt. After earning his Master's degree in mathematics from the University of Heidelberg, he joined the scientific staff of the data processing department of the German Cancer Research Centre in 1974. There he was involved with designing and implementing a general-purpose data base system, writing and supporting program-generating systems, supporting scientists using various application systems, and writing systems for them. His work at the Centre introduced him to APL.



In 1980, Gert worked at IBM's Scientific Centre in Heidelberg. Last year, he joined I. P. Sharp's Düsseldorf office and was acting branch manager. ■

## Melbourne

**Gary Cruth** has been appointed branch manager in Melbourne. He holds a Bachelor of Commerce degree in economics from the University of Melbourne.

Gary has 19 years of data processing experience, beginning his career as a specialist in the banking industry with IBM. Then he moved to Hong Kong as a senior consultant with Peat Marwick Mitchell. Later he was appointed manager in Victoria for Computer Sciences of Australia and then southern region manager for Rand McDonald Systems. For the past several years, Gary worked as an independent consultant specializing in strategic design studies. ■



## Melbourne

**Chris Read** has assumed responsibilities for applications development in Australia. In this newly created position, he is responsible for developing application software products for use by our customers worldwide.

Chris started his education in the United Kingdom and completed his B.Sc. and Ph.D. in mathematics at Monash University in Melbourne. He taught mathematics and physics in India. Just before joining I. P. Sharp, he was a lecturer in mathematics at the Royal Melbourne Institute of Technology.

Chris joined I. P. Sharp in 1979 to provide technical support for the Melbourne branch and was appointed branch manager later that year. ■

## Zurich

**Andreas Werder** has been appointed branch manager in Zurich. He started working in the Zurich office, on a part-time basis in 1978, while studying geography and social sciences at the University of Zurich.



After completing his graduate degree in geography, Andreas joined the Zurich office as an APL consultant. He has been involved in several projects on private data base and information systems. In 1982, he was on loan to the Paris office for six months.

He first started using APL in 1972 while attending college and has been a devoted user ever since. ■

---

## Publications

---

### Bookends

#### **SIFT Users' Guide** (*new*)

Forecasting is an integral part of both our business and personal lives. We want to know, for example, what the future holds for interest rates, housing prices, product sales, or the stock market.

SIFT, Sharp Interactive Forecasting Techniques, brings together a wide range of time series analysis techniques for forecasting a series. SIFT operates interactively, and prompts you for what it needs. The package at its elementary level is simple enough that you can obtain a forecast with just two basic commands. At a more complex level, it is sophisticated enough for you to try a number of techniques and compare their performances.

*October 1983, 140 pp., \$8.00*

## ... Bookends

### CAPERS Users' Guide (new)

Capital expenditure projects involve an initial outlay of large sums of money, with returns—in the form of revenue, or cost savings—spread over a number of years. Organizations need to know what their return on capital investment will be, before committing large sums to projects.

CAPERS, I.P. Sharp's Computer-Assisted Project Evaluation and Reporting System, is designed to help in the evaluation of a wide range of capital expenditure projects. You can use CAPERS to evaluate projects concerning plant expansion, equipment acquisition, product introduction, real estate development, energy development, and mining. In fact, you can use CAPERS to your advantage in any situation where cost/benefit analysis is needed.

August 1983, 41 pp., \$4.00

### USFLOW Data Base (new)

The U.S. Flow of Funds Quarterly data base contains over 3600 time series on the sources and uses of funds in the U.S. economy. The financial and non-financial activities for over

50 major economic sectors are discussed. Three facts are covered for the over 1200 accounts in the data base—period end stock, flow, and seasonally adjusted flow. The data is supplied by the U.S. Federal Reserve Board.  
July 1983, 37 pp., \$4.00

### Financial Post Corporate Data Base Manual (new)

The FPCORP data base consists of current and historical corporate statistics for approximately 470 Canadian corporations. Annual data includes balance sheet and income statement statistics compiled from corporate statements, plus calculated ratios such as profit margins, ratios of returns, and capital structure. There are also quarterly statistics, including revenue, tax, dividend, and per share information.

October 1983, 102 pp., \$7.00



## mailing request

- Please amend my mailing address as indicated.
- Please add the following name(s) to your Newsletter mailing list.
- Please send me a Publications Order Form.
- Please add my name to the Aviation Newsletter mailing list.
- Please add my name to the Energy Newsletter mailing list.
- Please add my name to the Financial and Economic Newsletter mailing list.
- Please add my name to the PROMIS Newsletter mailing list. PROMIS is a computer-aided manufacturing system for microelectronics.

Name: \_\_\_\_\_

Title: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Please send me information about your courses in:

City: \_\_\_\_\_ Tel.: \_\_\_\_\_

The I.P. Sharp Newsletter is published by I.P. Sharp Associates, Suite 1900, 2 First Canadian Place, Toronto, Canada M5X 1E3.  
Your comments and contributions are welcome.

Editor: Irene Shimoda  
Design and Layout: Susan Storey,  
Carol Rutherford  
Circulation: Mary Kopfensteiner

Printed in Canada  
November 1983  
ISSN 0226 854X

# international offices

## Aberdeen

I.P. Sharp Associates Limited  
5 Bon Accord Crescent  
Aberdeen AB1 2DH  
Scotland  
(0224) 25298

## Amsterdam

InterSystems BV  
Kabelweg 47  
1014 BA Amsterdam  
The Netherlands  
(020) 86 80 11  
Telex: 18795 ITS NL

## Atlanta

I.P. Sharp Associates, Inc.  
1210 S. Omni International  
Atlanta, Georgia 30335  
(404) 586-9600

## Boston

I.P. Sharp Associates, Inc.  
1 Liberty Square  
Boston, Massachusetts 02109  
(617) 542-2313

## Brisbane

I.P. Sharp Associates Pty. Ltd.  
6th Floor, Dalgety House  
79 Eagle Street  
Brisbane, Queensland 4000  
Australia  
(07) 229-8330

## Brussels

I.P. Sharp Europe SA  
Boulevard de la Cambre 36, bte 5  
1050 Bruxelles  
Belgium  
(02) 649 99 77

## Calgary

I.P. Sharp Associates Limited  
Suite 550, Bow Valley Square 4  
250-6th Avenue S.W.  
Calgary, Alberta T2P 3H7  
(403) 265-7730

## Canberra

I.P. Sharp Associates Pty. Ltd.  
16 National Circuit  
Barton, A.C.T. 2600  
Australia  
(062) 73-3700

## Chicago

I.P. Sharp Associates, Inc.  
Suite 3860  
55 West Monroe Street  
Chicago, Illinois 60603  
(312) 782-3177

## Copenhagen

I.P. Sharp ApS  
Østergade 24B  
1100 Copenhagen K  
Denmark  
(01) 11 24 34

## Coventry

I.P. Sharp Associates Limited  
7th Floor B Block  
Coventry Point, Market Way  
Coventry CV1 1EA England  
(0203) 56562

## Dallas

I.P. Sharp Associates, Inc.  
Suite 1148, Campbell Center  
8350 North Central Expressway  
Dallas, Texas 75206  
(214) 369-1131

## Denver

I.P. Sharp Associates, Inc.  
Suite 416  
5680 South Syracuse Circle  
Englewood, Colorado 80111  
(303) 741-4404

## Dublin

I.P. Sharp Associates Limited  
Segrave House  
Earisfort Terrace  
Dublin 2, Ireland  
(01) 763605

## Düsseldorf

I.P. Sharp GmbH  
Kaiserswertherstrasse 115  
4000 Düsseldorf 30  
West Germany  
(0211) 45 20 52

## Edmonton

I.P. Sharp Associates Limited  
2358 Principal Plaza  
10303 Jasper Avenue  
Edmonton, Alberta T5J 3N6  
(403) 428-6744

## Frankfurt

I.P. Sharp GmbH  
Rossertstrasse 15  
am Palmengarten  
6000 Frankfurt 1  
West Germany  
(0611) 72 47 73

## Hallifax

I.P. Sharp Associates Limited  
Suite 706, Cogswell Tower  
2000 Barrington Street  
Halifax, Nova Scotia  
B3J 3K1  
(902) 423-6251

## Helsinki

TMT-Team Oy (Agent)  
Westendintie 1  
SF-02160  
Espoo 16, Finland  
(0) 452 1155

## Hong Kong

I.P. Sharp Associates (HK) Limited  
Suite 606, Tower 1  
Admiralty Centre, Hong Kong  
5-294341

## Houston

I.P. Sharp Associates, Inc.  
Suite 375, One Corporate Square  
2600 Southwest Freeway  
Houston, Texas 77098  
(713) 526-5275

## London, Canada

I.P. Sharp Associates Limited  
Suite 510, 220 Dundas Street  
London, Ontario N6A 1H3  
(519) 673-4060

## London, England

(European Headquarters)  
I.P. Sharp Associates Limited  
132 Buckingham Palace Road  
London SW1W 9SA, England  
(01) 730-4567  
Telex: 8954178 SHARP G

## Los Angeles

I.P. Sharp Associates, Inc.  
Suite 1230  
1801 Century Park East  
Los Angeles, Ca. 90067  
(213) 277-3878

## Madrid

I.P. Sharp Associates Limited  
Serrano 23, Piso 8  
Madrid 1, Spain  
(91) 276 70 54

## Melbourne

I.P. Sharp Associates Pty. Ltd.  
520 Collins St., 7th Floor  
Melbourne, Victoria  
3000, Australia  
(03) 614-1766

## Mexico City

Teleinformatica de Mexico SA  
(Agent) Mail to:  
Arenal N 40, Chimalistac  
Mexico 20 D.F., Mexico  
(905) 550-8033

## Miami

I.P. Sharp Associates, Inc.  
Suite 240  
15327 N.W. 60th Avenue  
Miami Lakes, Florida 33014  
(305) 556-0577

## Milan

Informatical Society Italia Srl  
(Agent)  
Via Eustachi 11  
20129 Milan, Italy  
(02) 221 612

## Montreal

I.P. Sharp Associates Limited  
Suite 1610  
555 Dorchester Boulevard W.  
Montreal, Quebec H2Z 1B1  
(514) 866-4981

## New York

I.P. Sharp Associates, Inc.  
Suite 210  
230 Park Avenue  
New York, N.Y. 10169  
(212) 557-7900

## Newport Beach

I.P. Sharp Associates, Inc.  
Suite 1135  
610 Newport Center Drive  
Newport Beach, Ca. 92660  
(714) 644-5112

## Oslo

I.P. Sharp A/S  
Postboks 486 Sentrum  
Dronningens gate 34  
Oslo 1, Norway  
(02) 41 17 04

## Ottawa

I.P. Sharp Associates Limited  
Suite 600, 265 Carling Ave.  
Ottawa, Ontario K1S 2E1  
(613) 236-9942

## Palo Alto

I.P. Sharp Associates, Inc.  
Suite 201, 220 California Ave.  
Palo Alto, Ca. 94306-1683  
(415) 327-1700

## Paris

I.P. Sharp SARL  
9 Rue du Cirque  
75008 Paris  
France  
(1) 225 98 20

## Rochester

(United States Headquarters)  
I.P. Sharp Associates, Inc.  
1200 First Federal Plaza  
Rochester, N.Y. 14614  
(716) 546-7270

## Rome

Informatical Society Italia Srl  
(Agent)  
Piazza Della Rotonda 2  
00100 Rome, Italy  
(06) 656-5925

## San Francisco

I.P. Sharp Associates, Inc.  
Suite C-415, 900 North Point St.  
San Francisco, Ca. 94109  
(415) 673-4930

## Saskatoon

I.P. Sharp Associates Limited  
Suite 303, Financial Bldg.  
230-22nd St. E.  
Saskatoon, Saskatchewan  
S7K 0E9  
(306) 664-4480

## Seattle

I.P. Sharp Associates, Inc.  
Suite 223, Executive Plaza East  
12835 Bellevue Redmond Road  
Bellevue, Washington 98005  
(206) 453-1661

## Seoul

Daewoo Corporation (Agent)  
541, 5-Ga, Namdaemoon-Ro  
Jung-Gu (CPO Box 2810)  
8269 Seoul, Korea.  
771-91/2  
Telex: 23341-5/24295 DAEWOO K

## Singapore (Far East H.Q.)

I.P. Sharp Associates(S) Pte. Ltd.  
Suite 1601, CPF Building  
79 Robinson Road  
Singapore 0106  
Republic of Singapore  
2230211  
Telex: 20597 IPSAS RS

## Singapore

Singapore International  
Software Services Pte. Ltd.  
(Agent)  
Suite 1601, CPF Building  
79 Robinson Road  
Singapore 0106  
Republic of Singapore  
2230211

## Stockholm

I.P. Sharp AB  
Kungsgatan 65  
S111 22 Stockholm, Sweden  
(08) 21 10 19

## Sydney (Australian H.Q.)

I.P. Sharp Associates Pty. Ltd.  
8th Floor, Carlton Centre  
55 Elizabeth St.  
Sydney, New South Wales 2000  
Australia  
(02) 232-6366

## Tokyo

INTEC, Inc. (Agent)  
Dai 23 Mori Bldg. 4th Floor  
1-23-7 Toranomon  
Minato-ku, Tokyo 105  
Japan  
(03) 502 8781  
Telex: 2322008 INTECA J

## Toronto

(International Headquarters)  
I.P. Sharp Associates Limited  
Suite 1900  
2 First Canadian Place  
Toronto, Ontario  
M5X 1E3  
(416) 364-5361  
Telex: 0622259 I P SHARP TOR

## Vancouver

I.P. Sharp Associates Limited  
Suite 902, 700 West Pender St.  
Vancouver, B.C. V6C 1G8  
(604) 687-8991

## Victoria

I.P. Sharp Associates Limited  
Chancery Court  
1218 Langley Street  
Victoria, B.C. V8W 1W2  
(604) 388-6365

## Vienna

I.P. Sharp Ges.mbh  
Renngasse 4  
A-1010 Wien, Austria  
(0222) 66 42 48

## Warrington

I.P. Sharp Associates Limited  
1-3 Dolmans Lane  
Warrington, Cheshire  
WA1 2ED England  
(0925) 50413/4

## Washington

I.P. Sharp Associates, Inc.  
Suite 305, 2033 K Street N.W.  
Washington, D.C. 20006  
(202) 293-2915

## Wayne

I.P. Sharp Associates, Inc.  
Suite 303  
155 Willowbrook Blvd.  
Wayne, New Jersey 07470  
(201) 785-8050

## White Plains

I.P. Sharp Associates, Inc.  
Suite 312 West  
701 Westchester Avenue  
White Plains, New York 10604  
(914) 328-8520

## Winnipeg

I.P. Sharp Associates Limited  
Suite 208  
213 Notre Dame Avenue  
Winnipeg, Manitoba R3B 1N3  
(204) 947-1241

## Zurich

I.P. Sharp AG  
Fortunagasse 15  
CH-8001 Zurich, Switzerland  
(01) 211 84 24

## I. P. Sharp Network

APL OPERATOR VOICE 416 363-2051 COMMUNICATIONS 416 363-1832

Our private, packet-switched network connects with the Value Added Networks—Datapac, Datex-P, PSS, Telenet, Telepac, Transpac, and Tymnet—to provide access from the 46 countries listed below:

- Argentina • Australia • Austria • Bahamas • Bahrain • Barbados • Belgium • Bermuda • Brazil • Canada • Chile • Denmark
- Dominican Republic • Finland • France • French Antilles • Germany • Great Britain • Hong Kong • Hungary • Ireland
- Israel • Italy • Japan • Korea • Kuwait • Luxembourg • Mexico • The Netherlands • New Zealand • Norway • The Philippines
- Portugal • Puerto Rico • Qatar • Republic of Singapore • Republic of South Africa • Saudi Arabia • Scotland • Spain
- Sweden • Switzerland • Taiwan • United Arab Emirates • U.S.A. • Virgin Islands

SHARP APL is accessible from over 600 places via a local telephone call. Please ask your nearest I.P. Sharp office or representative for a complete list of access points and access procedures. Our private network also connects with the worldwide Telex network via the Amsterdam and Rochester, New York nodes.