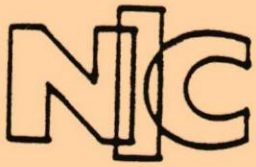


PC SAM - Simple Access to E-MAIL

WHAT IS SAM?

SAM is a communication and mail management software package that makes electronic mail (e-mail) fast, efficient, and easy to access from a PC. E-mail is the most used and useful service on the DDN. SAM can help you get the maximum benefit from DDN e-mail, even if you work on a heavily loaded host.

At the push of a key, SAM can connect to the mail host, retrieve any messages that have been delivered to you, and move them easily to your PC. Using SAM, you can read and compose mail on your PC, thus saving connect time and avoiding the frustrations of working on a slow host computer. When you've finish writing outgoing messages on your PC, SAM can automatically send them to the mail host for online delivery.

SAM includes a built-in editor for creating messages and editing DOS text files. Or, if you prefer, you can create messages using any PC word processor that can create ASCII text files, and SAM can pull those files into your message.

SIMPLICITY

Sam is easy for beginners to use, yet provides enough features and convenience to satisfy experienced users.

Function keys control SAM, making it simple to learn. A display of the function keys, each labelled with its current function, is always at your fingertips for quick reference.

The screen looks and feels like a desk, with IN, OUT, and other file baskets to hold your documents. Messages can be copied, moved, filed, or discarded, just like dealing with paper.

Several special features make network use even easier. The "Address Book" holds the e-mail addresses your use most frequently, and lets them be easily copied to the "To:" field of messages without retyping. E-mail and other DDN services can be accessed at the press of a key.

There are 16 empty baskets which you can label and use for sorting your mail. When you no longer need a document, just move it into SAM's WASTE basket for disposal.

SAM handles many of the network access procedures automatically. It can login to TACs, make host connections, and reach across the network to use WHOIS, TACNEWS and BIBLIO services, all at the touch of a single key.

A special "Learn Mode" captures and stores an entire login sequence, letting you create new services to meet your needs. "Learn" can also be used by experienced SAM users to create custom function keys.

POWER

SAM's Macro language can be used to adapt the connection procedure for a variety of network hookup configurations. SAM can go through LANs, switches, modems... whatever you use to connect to the DDN. Macro commands can be attached to the function keys of each service, to automate frequently repeated online tasks.

SAM emulates the popular VT100 terminal, to facilitate working directly with you host. The interactive service also provides KERMIT and XMODEM file transfer capabilities, for moving files between your PC and host computer.

REQUIREMENTS

SAM requires an IBM PC (or compatible) computer with at least 256K of RAM memory (320K to use KERMIT) and two disk drives. For maximum useability, a hard disk and 640K memory are recommended. Color graphics are supported, but not required. SAM is not copy-protected. SAM currently supports the following host mail programs: UNIX (BSD 4.x) mail, VMS mail, InfoMail, and MM.

SAM has been tested by DARPA, NSF, and the DDN PMO.

COST AND SITE LICENSING FOR SAM

I. ORDERING SAM

SAM can be ordered from the NIC by sending a check, money order or purchase order for the total amount of the order in US dollars, made payable to SRI International. Include with all orders your full name, US mailing address with zip code, telephone number and network mailbox (if available) and send to:

DDN Network Information Center
SRI International, Room EJ291
333 Ravenswood Avenue
Menlo Park, CA 94025

Each order should state whether the purchase is for one or more individual copies, or for a site license (master disk), and should indicate how many units are being ordered. The order should also state clearly the name of the mail program to which SAM will interface and the host on which that mail program resides.

For more ordering information call (800) 235-3155 or (415) 859-3695.

II. TRIAL DISKS

There will be a limited number of SAM trial disks available for $10.00 each so that users can try the program on a trial basis. Requests for the trial disk should be directed to the address listed above.

III. SITE LICENSING

Most of the military and government organizations with whom we have talked have requested a site-license marketing arrangement for SAM to meet their budgeting requirements. Therefore, two kinds of site licensing are being offered:

1. A simple discount for volume purchase.
2. A site license consisting of a master disk from which a set number of copies of the program can be made, and from which copies of the user manual can be made. Each copy of the program will be serialized so that once the agreed upon number of copies has been made, the master copy will no longer be valid.

Cost details for both arrangements are outlined in the Pricing section below.

IV. PRICING

1. SINGLE UNIT PRICE

   The single-unit price for the SAM program and manual is $150 from SRI International.

2. VOLUME DISCOUNT PRICE

NUMBER OF UNITS	$ COST/UNIT (w/ manual)
2-9	145
10-49	125
50-99	100
100-499	90
500-999	85
1001 +	75

SITE LICENSING PRICE

NUMBER OF UNITS	$ COST/SITE LICENSE *
25 units**	3,000
50 units	4,750
100 units	8,500
500 units	40,000
1000 units	70,000

*Includes all software upgrades for one year and the text of the manual on a floppy disk plus 1 camera-ready hardcopy with full illustrations.

** Minimum

3. SHIPPING & HANDLING

NUMBER OF UNITS	$ SHIPPING/HANDLING
1	3.00
2-10	5.00
11-20	10.00
over 20	.40/unit*** + 5.00 (flat fee)

***For Example:

25 units:	(.40 x 25) + $5.00 = $15.00
40 units:	(.40 x 40) + $5.00 = $21.00
75 units:	(.40 x 75) + $5.00 = $35.00

For orders of 100 units or more, please call for quotation.

Site license agreements - flat fee of $5.00 (any quantity).

All orders are shipped UPS Ground Service, however shipment by Next-Day Air or 2nd-Day Air is available (call for price quotation). Please allow 4-6 weeks delivery.

4. TAX

California Residents please add 6.5% sales tax.

VI. CUSTOMER SUPPORT

Comments and questions can be sent by registered SAM users to the online mailbox, SAM@SRI-NIC.ARPA.

SRI will maintain a list of registered SAM users. Only registered users will be eligible for online customer support and warranty. Purchase of a site license for SAM includes any upgrades for the following year. Upgrades will either be made by downloading change files via the network or by issuing a floppy disk.

V. TRAINING

No training is included in the purchase price of SAM. Training can be negotiated separately with SRI International, if needed. Rates are $500/day plus expenses on a consulting basis, or $200 per attendee on a classroom basis, plus expenses if the class is held at a facility other than SRI International, Menlo Park, CA.

VI. SOFTWARE ENHANCEMENTS

SAM is marketed "as is," to work with standard versions of the supported electronic mail systems. Customization is possible and can be negotiated on a case-by-case basis at an additional cost.

NIC 50008

Dennett

PC SAM
Simple Access to e-Mail
User's Guide

SEPTEMBER 1986

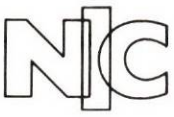
PC SAM
Simple Access to e-Mail
User's Guide

September 1986

Based on PC COMplete, a product of
Transend Corporation

Edited by:
Stephen C. Dennett

Published by:



DDN Network Information Center
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025

IMPORTANT NOTICE

* WARNING *

Printed and bound in the United States of America. Published by the DDN Network Information Center, SRI International, Menlo Park, CA 94025.

TABLE OF CONTENTS

List of Figures

SECTION 1. INTRODUCTION

1.1 Introduction

Simple Access to e-Mail (SAM) on your personal computer (PC)[1] makes it easy to use a variety of message systems available on the Defense Data Network (DDN). With a single command you can send and receive quick messages, long documents, or any other text files.

Beginners find SAM easy to learn. Occasional users like not having to relearn SAM every time they want to use it. Frequent communicators like SAM's ease-of-use and powerful features.

SAM is set up like a well-organized desk. There is an IN basket for incoming messages, an OUT basket for outgoing messages, an ADDRESS BOOK for often-used addresses, and even a WASTE basket.

How do you send a message? Just type it in, put it in the OUT basket, and press a single key. SAM does the rest. First it connects to the host computer where your electronic mailbox is. If you normally go through a Terminal Access Controller (TAC), SAM will call the TAC, log in, and connect through the TAC to your mail host. Once connected to your mailbox, SAM transmits all the messages in your OUT box to the host mail program, which sends the messages through the DDN to their recipients. SAM also picks up messages waiting for you in your mailbox and stores them in your personal computer.

You can also show SAM how to call up and log into other computers or database services. SAM records how you do it the first time, and after that does it for you.

Finally, SAM provides tools for easily managing your DOS files. At the push of a button you can move, delete, or rename files, and SAM's editor can be used to compose messages and create text files.

1.2 Features

1. Automated communication with network mail:
    a. Dials the DDN (with autodial modem)
    b. Logs in with your personal identifier and network access codes
    c. Connects to the host on which you maintain a DDN Mailbox
    d. Automatically logs into your Mailbox Account
    e. Sends messages
    f. Receives messages
2. Single-keystroke commands, with on-screen menu
3. Easy-to-use Desktop display, including:
    a. IN basket for newly received messages
    b. OUT basket for messages to be sent
    c. ADDRESS BOOK for recipients' addresses
    d. SERVICES and YOUR PC baskets for communication information
    e. WASTE basket for discarded messages
    f. 16 baskets for filing messages
4. Lets you write and revise messages, using a built-in text editor

[1]SAM currently runs only on IBM-compatiple personal computers.

5. Displays and prints messages
6. Terminal emulation, including VT-100

1.3 How To Use This Manual

This users guide gives you simple step-by-step instructions on setting up and using SAM. The instructions are written for someone who has little or no computer experience.

Sections 1 through 8 introduce SAM, and show how to install, configure and use the primary features of the program. They describe how to use SAM's basic functions, and are designed to provide all of the information many SAM users will ever need.

Sections 9 through 16 discuss SAM's advanced features, and how to use them. Each of these sections stands alone. Feel free to read only those sections that interest you.

If you have just received SAM, begin with Section 2, Getting Started. If SAM has already been set up for your computer, begin with Section 3, Starting SAM.

This guide was written for the IBM Personal Computers. If you are using an IBM-compatible computer, some things may not appear exactly as shown in this guide.

1.4 User Registration

Becoming a registered SAM user will entitle you to customer support and information on new releases. SAM User registration is done by electronic mail. After reading the first five chapters of this manual, simply fill out the form contained in the REGISTRATION box (on SAM's desktop). Then use SAM to send it in.

1.5 A Note on Kermit

Your SAM disks include the IBM PC version (2.29) of KERMIT, a communications program and file transfer protocol, along with the KERMIT manual. When SAM is installed, Kermit will also be copied to your floppy or hard disk.

The KERMIT file transfer protocol was develped at the Columbia University Center for Computing Activities. KERMIT is available for many systems for only a nominal fee from Columbia and from various user group organizations, such as DECUS and SHARE. The price of SAM was not increased by the inclusion of KERMIT.

KERMIT is free and available to anyone who requests it; however, it is not in the "public domain". The protocol, the manuals, the Columbia implementations, and many of the contributed implementations are copyrighted as follows:

> Copyright (C) 1985, Trustees of Columbia University in the City of New York. Permission is granted to any individual or institution to use, copy, or redistribute this software so long as it is not sold for profit, provided this copyright notice is retained.

For more information on KERMIT, contact:

> KERMIT Distribution
> Columbia University Center for Computing Activities
> 612 West 115th Street
> New York, NY 10025

Note that although we have included KERMIT for your convenience, SRI does not maintain KERMIT. Questions relating to the program should be directed to Columbia University[2].

[2]Send KERMIT questions online to the DDN mail address INFO-KERMIT@CU20B.

SECTION 2. GETTING STARTED

2.1 Getting started

This section will help you to get SAM running on your IBM PC. The installation process is fairly long, but need only be done once. After that you will be able to run SAM by just pressing a few keys.

There are two main steps:

1. Installation--preparing your PC and the SAM program for use.
2. Configuration--customizing SAM for your personal use.

You will need:

- This User Guide.
- The "SAM Installation" and "SAM Services" diskettes.
- A DDN mail user name and password
- Knowlege of which of the mail programs listed below you are using:
    1. MM
    2. Infomail
    3. VMS Mail
    4. UNIX mail[3]
- A TAC User ID and access code (if you use a TAC to access your host), authorized by your Host Administrator and issued by the DDN Network Information Center (NIC).
- The "network address" of the DDN host computer on which you maintain a mailbox.
- The account name and password required to log on to the host computer.
- In some cases, a separate mail system user name and mail system password.
- An IBM PC, XT, AT or compatible computer with at least 256K[4] of memory, and either:
    - Two floppy diskette drives or,
    - A hard disk and at least one floppy diskette drive.[5]
- A modem.
- A printer (optional).
- If your PC does not have a hard disk, you will also need:
    - A DOS (Disk Operating System) diskette, Version 2.0 or later
    - Three blank diskettes

This section has installation instructions for PCs with at least two floppy diskette drives and PCs with a hard disk and at least one floppy diskette drive. Use the installation instructions that are appropriate for your equipment. Once you have installed SAM, continue to the configuration instructions.

---

[3]If using a UNIX mail program, you will need to know if you have an "rc" options file. If you aren't certain, check with your local Host Administrator.

[4]SAM requires 256k to run, 320k to be able to execute DOS commands from within SAM or to use KERMIT.

[5]A hard disk is strongly recommended for maximum utility and efficiency.

2.2 Installation--PC With Two Diskette Drives

Before you start, make sure that you have a copy of the DOS diskette, Version 2.0 or later. Then:

1. **Insert the DOS diskette in drive A.** Drive A is usually the left or upper drive. When inserting, hold the diskette by its label, with the label facing up.
2. **Turn on your PC.** If necessary, turn on the video display monitor.
3. **If the PC asks you to, set the date and time.** If your PC has a built-in clock, you will not be asked to set the date and time, and you should skip this step. The PC will ask first for the date, and then for the time. Provide them accurately, since SAM marks them on your messages. Type in the date, and then press the RETURN key. Then type in the time using a 24-hour clock, e.g. enter 4:17 p.m. as 16:17, and press the RETURN key.

   If you realize that you made a mistake, type:

   date

   and press the RETURN key to reenter the date, or type:

   time

   and press the RETURN key to reenter the time.

Next, get the three blank diskettes mentioned earlier in this section.
Label them:

SAM Program Copy

SAM Messages

SAM KERMIT

Now format the diskettes, putting DOS on the SAM Program disk if you want it to be usable as a startup disk.[6] If you don't know how to format disks, follow these step-by-step instructions:

1. **After the A> prompt, type:**

   format b:\s

   and press the RETURN key.
2. **Insert the diskette labeled "SAM Program Copy" in drive B.** Drive B is usually the right-hand or lower drive.
3. **Press the RETURN key again.** When the computer asks if you want to format another diskette,
   type:

   n

   and press the RETURN key.
4. **Remove the SAM Program Copy diskette from drive B.**
5. **Insert the diskette labeled "Messages" in drive B.**
6. **After the A> prompt, type: format b:**

   and press the RETURN key.

---

[6]If you don't mind using a separate DOS startup disk, leaving DOS off the SAM PROGRAM disk will give you more storage space for messages.

7. **Press the RETURN key again.** When the computer asks if you want to format another diskette,
   type:

   y

   and press the RETURN key.
8. **Remove the SAM Messages diskette from drive B.**
9. **Insert the SAM KERMIT diskette in drive B.**
10. **Press the RETURN key again.** When the computer asks if you want to format another diskette,
    type:

    n

    and press the RETURN key.

Now you are going to install SAM on your PC. You should see the A> prompt. Then:

1. **Insert the "SAM Installation" diskette in drive A, and the "SAM Program Copy" diskette in drive B.**
2. **At the A> prompt, type:**
3. **install**

   and press the RETURN key. SAM installation instructions will appear.
4. **At the A> prompt,**

If your mail program is:	Type:
MM	**i-pc mm**
InfoMail	**i-pc infomail**
UNIX Mail	**i-pc unixmail**
VMS mail	**i-pc vms**

   Then press the RETURN key. Follow the directions on the screen, and press any key when ready.

5. **Replace the "SAM Program Copy" diskette with the "Messages" diskette when you are prompted to do so.** Then press any key.
6. **Check the screen for any error messages.** If you see any, consult your system support staff. If there are no errors, the first part of the installation was successful. Press any key.
7. **Further instructions will be displayed.**
8. **Remove the "SAM Installation" diskette from drive A.**
9. **Place the "SAM Services" diskette in drive A.** Leave the "Messages" diskette in drive B.
10. **At the A> prompt, type:**

    install2

    and press the RETURN key. Read the SAM installation instructions which appear.
11. **At the A> prompt,**

If your mail program is:	Type:
MM	**i-pc2 mm**
InfoMail	**i-pc2 infomail**
UNIX Mail	**i-pc2 unixmail**
VMS mail	**i-pc2 vms**

Then press the RETURN key. Follow the directions on the screen, and press any key when ready.

12. **Leave the MESSAGES diskette in drive B.** You will now see instructions on installing KERMIT.
13. **At the A> prompt, type:**

    i-pckerm

    and press the RETURN key. More SAM installation instructions will appear.
14. **When directed to do so, place the SAM KERMIT disk in drive B, then press any key to continue the installation.**
15. **When the installation is complete, remove the "SAM Services" diskette from drive A.** Store it and the "SAM Installation" disketted in a safe place. From now on, you will need only the "SAM Program Copy" and "Message" diskettes.
16. **Insert the SAM Program Copy diskette in drive A.**
17. **After the A> prompt, type:**

    sam

    and press the RETURN key. You have just installed and started SAM.

2.3 Installation--PC With A Hard Disk Drive

Your hard disk must be initialized with DOS 2.0 or later. If it is not, see the IBM DOS manual, or ask the system administrator at your organization for assistance. Then:

1. **Turn on your PC.** The switch is on the right-hand side. If necessary, turn on the monitor.
2. **If the PC asks you to, set the date and time.** If your PC has a built-in clock, you will not be asked to set the date and time, and you should skip the next step.
3. The PC will ask first for the date, and then for the time. Provide them accurately, since SAM marks them on your messages. Type in the date, and then press the RETURN key. Then type in the time using a 24-hour clock (e.g. enter 4:17 p.m. as 16:17), and press the RETURN key.

   If realize that you made a mistake, type:

   date

   and press the RETURN key to reenter the date, or type:

   time

   and press the RETURN key to reenter the time.

Now you are going to install SAM on your PC. You should see the C> prompt. Then:

1. **Insert the "SAM Installation" diskette in drive A.**
2. **Type:**

   a:

   and press the RETURN key. Be sure to include the colon.
3. **At the A> prompt, type:**

   install

   and press the RETURN key. Some installation instructions will appear, followed by another A> prompt.
4. **Note which drive your hard disk is (usually C), and what you want to call the subdirectory where SAM's files will reside.** In the example below, "c:" is used for the hard disk drive, and "subdirectory" where you will type in the name of the subdirectory you wish SAM to create. This subdirectory should not be one that already exists.
5. **At the A> prompt,**

If your mail program is:	Type:
MM	**i-xt c:\subdirectory mm**
InfoMail	**i-xt c:\subdirectory infomail**
UNIX Mail	**i-xt c:\subdirectory unixmail**
VMS mail	**i-xt c:\subdirectory vms**

   Then press the RETURN key. The PC will ask for a "command name". The command name you choose is what you will type in the future to start SAM (e.g., "mail", "sam", etc.). Type your choice (up to 8 letters), and press the RETURN key.
6. **Check the screen for any error messages.** If you see any, consult your system support staff. If there are no errors, the first part of the installation was successful. Press any key.
7. **Remove the "SAM Installation" diskette from drive A.**
8. **Place the "SAM Services" diskette in drive A.** Leave the "Messages" diskette in drive B.

9. **At the A> prompt, type:**

   install2

   and press the RETURN key. Read the SAM installation instructions which appear.

10. **At the A> prompt,**

If your mail program is:	Type:
MM	**i-xt2 c:\subdirectory mm**
InfoMail	**i-xt2 c:\subdirectory infomail**
UNIX Mail	**i-xt2 c:\subdirectory unixmail**
VMS mail	**i-xt2 c:\subdirectory vms**

   Then press the RETURN key. Follow the directions on the screen, and press any key when ready. After the files are copied, instructions for installing KERMIT will appear that say:

> Now you will finish installing SAM by adding KERMIT.
> Decide what subdirectory you wish to use to work on
> files. Create that subdirectory on your hard disk if it
> doesn't already exist.

11. **Decide what subdirectory you want SAM to use for storing information captured off the screen, files downloaded, or text files you create. Create this subdirectory if it doesn't already exist.** Note that this is not the subdirectory you specified earlier for SAM to keep its own files in. In the next step, "xxxxx" is the name of the "Work on Files" subdirectory.

12. **At the A> prompt, type:**

    i-xtkerm c:\xxxxx

    and press the RETURN key. More SAM installation instructions will appear.

13. **When the installation is complete, remove the "SAM Services" diskette from drive A.** Store it and the "SAM Installation" diskette in a safe place.

14. **At the C> prompt, type the command name you chose in step 4.**

    You have just installed and started SAM.

2.3.1 Installing Multiple SAMs on a Hard Disk

If your PC is used by more than one person, you may have several copies of SAM installed on its hard disk. Each copy of the SAM program normally requires about 300k of disk space before you begin saving messages. However, by altering your installation as described below you can run several copies of SAM with a single SAM program file. This will save about 200K per each copy of SAM for each copy installed after the first.

Note: If you aren't familiar with PC-DOS and batch files, you may need to consult your DOS manuals or get assistance from someone who has such familiarity.

To change the installation, you must edit the batch file that starts SAM, and then copy TX.EXE into your root directory. To do this:

1. **Turn on your PC and move to the hard disk root directory (usually C:\).**
2. **Edit the batch file that starts SAM.** During SAM installation you were asked what command you wanted to use to start SAM. This command is now the name of a batch file. (For example, if you replied that you wanted to start SAM with the command "mail", there

is now a file called MAIL.BAT in the root directory. Using the text editor of your choice (or SAM's editor; see Section 11.2), load this batch file for editing.

In this example, we will assume that your batch file is called MAIL.BAT and that you installed SAM in a subdirectory called "C:/SAM". MAIL.BAT will initially look like this:

```
echo off
c:
cd \sam
sam %1
```

3. **Add "c:\" to the beginning of the last line**, so the file looks like this:

```
echo off
c:
cd \sam
c:\sam %1
```

4. **Copy the program TX.EXE from your SAM subdirectory to your hard disk root directory (usually C:\), then delete TX.EXE from the SAM subdirectory.**
5. **For each additional SAM installation, modify the batch file in the manner described above, then delete TX.EXE from the subdirectory that installation uses for SAM.**

2.4 Configuring SAM

"Configuring" SAM means customizing it to work properly with your PC, modem, and printer. All you must do to configure SAM is fill out a few forms.

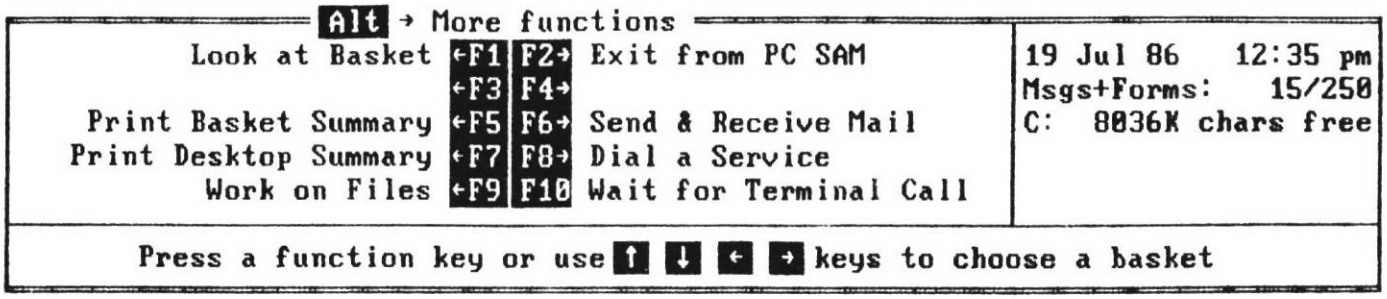
Before you configure SAM, look at Figure 2-1, the Desktop display. The Desktop is divided into four parts, called "windows":

- The Function Keys window
- The Status window
- The Instruction window
- The Baskets window

The information in each of the windows will vary from what you see here, depending on what SAM is doing.

Start SAM, if you have not already done so. You should see the Desktop display. Then:

1. **Move the highlighted bar to the YOUR PC basket.** Use the arrow keys on the numeric keypad.
2. **Press the F1 key (Look at Basket.)** The screen will show the contents of the YOUR PC basket.



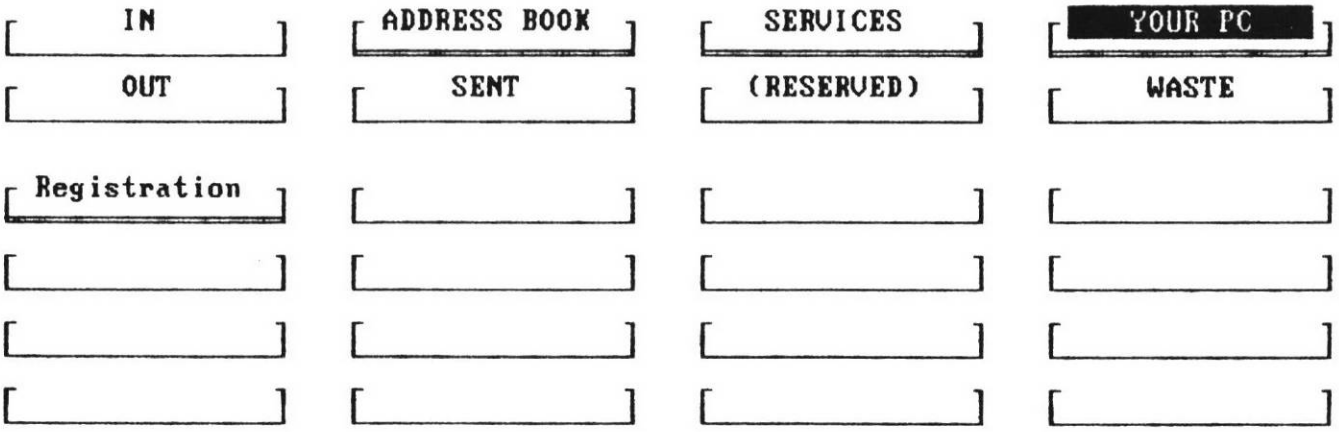


Figure 2-1: Desktop Display

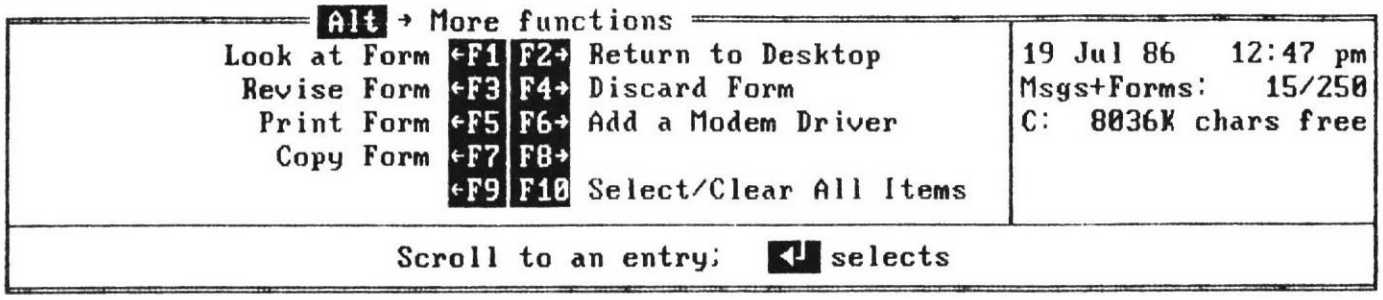
2.5 YOUR PC Basket

The YOUR PC basket contains these forms (see Figure 2-2):

- The "PC Configuration" form tells SAM about your computer, printer, and modem.
- The "Preferences" form lets you customize some of SAM's features to suit your preferences.
- Modem driver forms tell SAM how to use different kinds of modems.

The first form you should fill out is the "PC Configuration" form.

1. **Move the highlighted bar to the "PC Configuration" line.** Use the arrow keys on the numeric keypad.
2. **Press the F1 key (Look at Form.)** You will see the PC Configuration form.



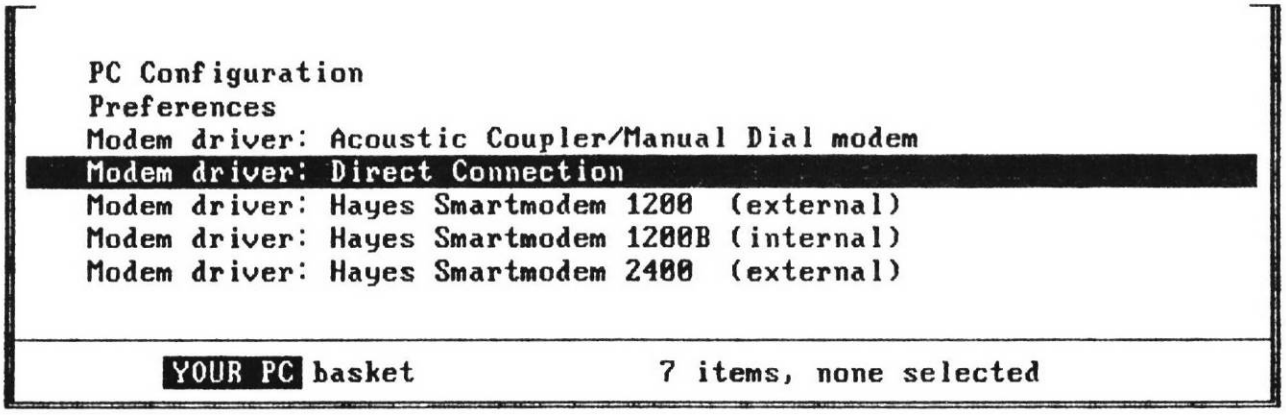
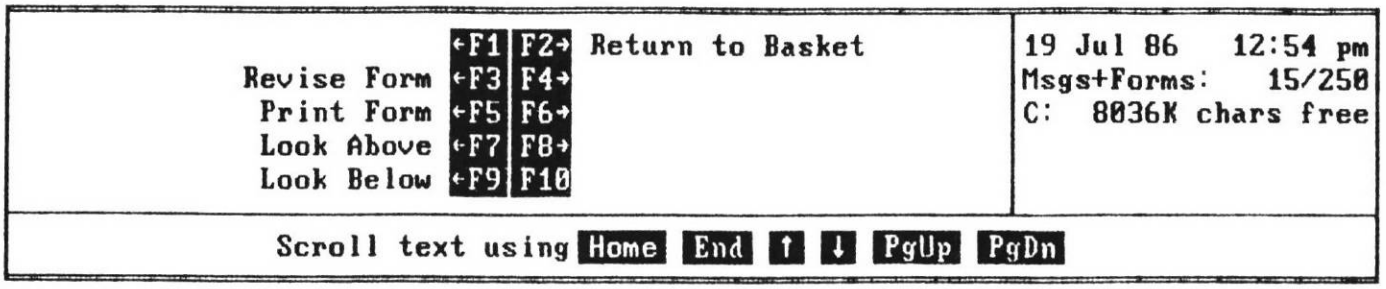


Figure 2-2: Your PC Basket

2.5.1 PC Configuration Form

The PC Configuration form contains information about your computer, printer and modem. See Figure 2-3.



P C C O N F I G U R A T I O N

What modem are you using?
=> Hayes Smartmodem 1200B (internal)

Which serial port is your modem attached to?
=> COM1 (usual choice)
COM2

What type of phone line do you have?
=> Touch Tone dialing
Rotary (pulse) dialing

Figure 2-3: PC Configuration Form

The "PC Configuration" form will already be filled out. All you have to do is make any necessary changes. Follow these instructions:

1. **Press the F3 key (Revise Form).** The first question is, "What modem are you using?" Use the up- and down-arrow keys to move the highlighted bar to your choice, then press the RETURN key.

   An arrow marks your choice. If you change your mind, reposition the bar and press the RETURN key again.

   NOTE: If you are using a modem compatible with the Hayes modem, we suggest that you select HAYES 1200 (external).

2. **Press the PgDn key.** The next question is, "Which serial port is your modem attached to?" Move the highlighted bar to your choice and press the RETURN key. On most PCs, the modem is attached to COM1.

3. **Press the PgDn key.** The next question is, "What type of phone line do you have?" Move the highlighted bar to your choice and press the RETURN key.

4. **Press the PgDn key.** The next question is, "Where is your printer connected?" If your printer is connected to a serial port, four more questions will appear. Answer each in the same manner as previous questions. (On most PCs, parallel printers are attached to LPT1, and serial printers are attached to COM2.)

5. **Press the PgDn key.** The last question is, "Disk\directory to be used for Work on Files or Online buffering." Since you will need to type in the answer, there is a blinking block cursor instead of a highlighted bar.

   If your PC has two diskette drives, type:

   a:

   If your PC has a hard disk, you should give the name of the subdirectory you used when installing KERMIT. Type:

   c:\xxxxx

   with "xxxxx" being the name of the subdirectory you gave when installing KERMIT.

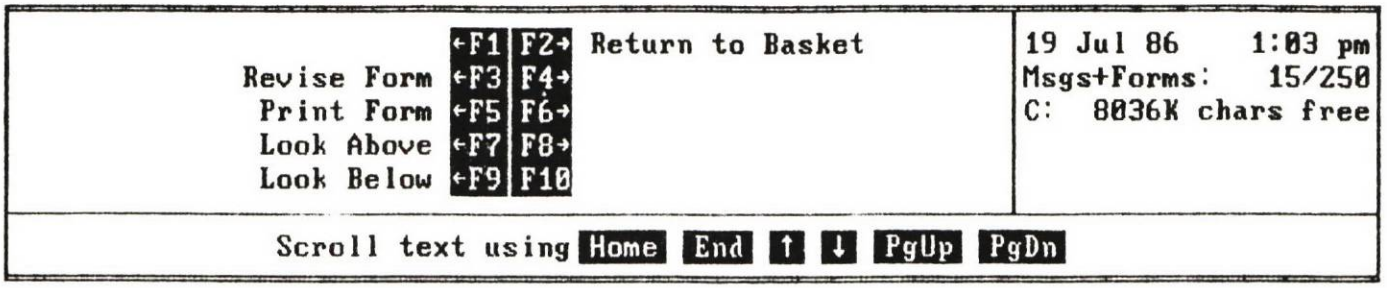
If you make a mistake or change your mind about an answer, you can always reenter it. Use the arrow keys and the Home, End, PgUp and PgDn keys to move the highlighted bar or blinking cursor to the mistake. Then enter the correct answer.

If you want to get rid of all of the changes you have made, press the F4 key (Restore Original).

Once you are done filling out the form, press the F2 key (Return Form to Basket). SAM will save your answers. Then, the screen will show the contents of the YOUR PC basket.

2.5.2 "Preferences" Form

The second form you should fill out is the "Preferences" form. The "Preferences" form controls some of the inner workings of SAM. See Figure 2-4.



P R E F E R E N C E S

Do you want sound?
⇒ Yes
No

How do you want the WASTE Basket emptied?
⇒ Leave it to be emptied manually (recommended)
Limit it to a specific size

Do you want to protect your Desktop with a password?
Yes
⇒ No

Figure 2-4: "Preferences" Form

To fill out the "Preferences" form, the cursor should be in the YOUR PC basket. For each question, Move the highlighted bar to your choice and press the RETURN key to select your answer.

1. **Move the highlighted bar to the "Preferences" line.**
2. **Press the F3 key (Revise form.)** The first question is, "Do you want sound?" If you say "yes", your PC will beep if you make a mistake.
3. **Press the PgDn key.** The next question is, "How do you want the WASTE Basket emptied?" Select whether you want your waste basket emptied manually or automatically after a certain number of messages have accumulated.
4. **Press the PgDn key.** "Do you want to protect your Desktop with a password?" If you want SAM to require you to enter a secret password each time it is started, and each time you end a mailing or online session, select YES.

   The password may have up to 12 characters. Type in the same password in both spaces.

   Be sure to record the password; you will not be able to use SAM without it. Note that

your password is also required to remove SAM's password protection, or to change the password.

We will refer to this password as the "SAM password". Do not confuse it with your TAC access code, which must be entered before using the DDN, or your host system password, which must be entered before accessing your host mailbox system.

5. **Press the PgDn key.** This question lets you set which basket SAM will go to after moving or copying a message. You will probably want to use SAM for a while before deciding which option to choose.
6. **Press the PgDn key.**"Are you using a color or monochrome display?" If you are using a color display, choose "color". When you select color, a list of choices will appear below the question. You can choose the colors for foreground, background, bright foreground, and error messages.

   Your color selections will not take effect until you return the preferences form to the basket. At that point, if the colors you have chosen are unreadable or otherwise undesirable, you can reset them by pressing the ALT and F4 keys simultaneously.

If you make a mistake or change your mind about an answer, you can always reenter it. If you want to get rid of the changes you have made, press the F4 key (Restore Original).

Once you are done filling out the form, press the F2 key (Return Form to Basket). SAM will save your answers. Then, the screen will show the contents of the YOUR PC basket.

2.6 SERVICES Basket

The next form that you should fill out is in the SERVICES basket. The SERVICES basket holds "service forms", which contain information that SAM uses to communicate with the DDN and to use your mailbox account. Once a service is filled out, it can run a series of commands to automatically connect SAM to your host. The "DDN Automatic Mail service" is a special service form programmed to automatically upload and download mail between SAM and your mail host.

In this manual, when we refer to "DDN Automatic Mail service", we are referring to the service form in SAM, not the DDN mail program that runs on your host. We will refer to the mail program that runs on your host and sends mail over the DDN as simply "DDN Mail".[7]

You will also notice four other forms in the SERVICES basket. Three of them, WHOIS, TACNEWS and BIBLIO, are DDN Network Information Center services. These Service forms are already completed, and need not be modified to use. The third form, INTERACTIVE TERMINAL, should be completed with the same information you put in the DDN Automatic Mail service.

To get to the SERVICES basket, follow these instructions:

1. **If you are still in the YOUR PC basket, press the F2 key (Return to Desktop).**
2. **Move the highlighted bar to the SERVICES basket.**
3. **Press the F1 key (Look at Basket).** You will see the contents of the SERVICES basket.

2.6.1 Network Forms

To reach the your DDN Mail program, SAM usually dials a Terminal Access Controller (TAC) on the DDN, which then connects you to your mail host. (You can adapt SAM to reach your mail host by other methods. These methods depend on the communication system used in your office.) Network forms will always appear at the bottom of the SERVICES basket, below regular service forms.

If you use the DDN to access electronic mail, read the following example, which shows how to fill out the DDN Network form.

Before filling out the Network form, you must:

- Obtain a TAC User ID and access code. These are authorized by the Host Administrator on your mail host, and are distributed by the DDN Network Information Center (NIC).
- Obtain the telephone number of the nearest TAC.

If you do not already have this information, call the NIC Hotline at (800) 235-3155 to obtain it.

More information about network forms is provided in Section 13.

2.6.2 Filling Out the DDN Network Form

To fill out the DDN network form, be sure you are in the "Services" basket. Then move the highlighted bar to "Network: DDN" and:

1. **Press the F3 key (Revise Service).** The DDN Network form will appear.
2. **Network Name.** The first item already displays the network name: DDN. Press the PgDn key to reach the next question.
3. **Enter the DDN phone number.** A blinking underline cursor will appear after this

[7]Your specific DDN Mail program may be called Infomail, MM, UNIX mail, or VMS mail.

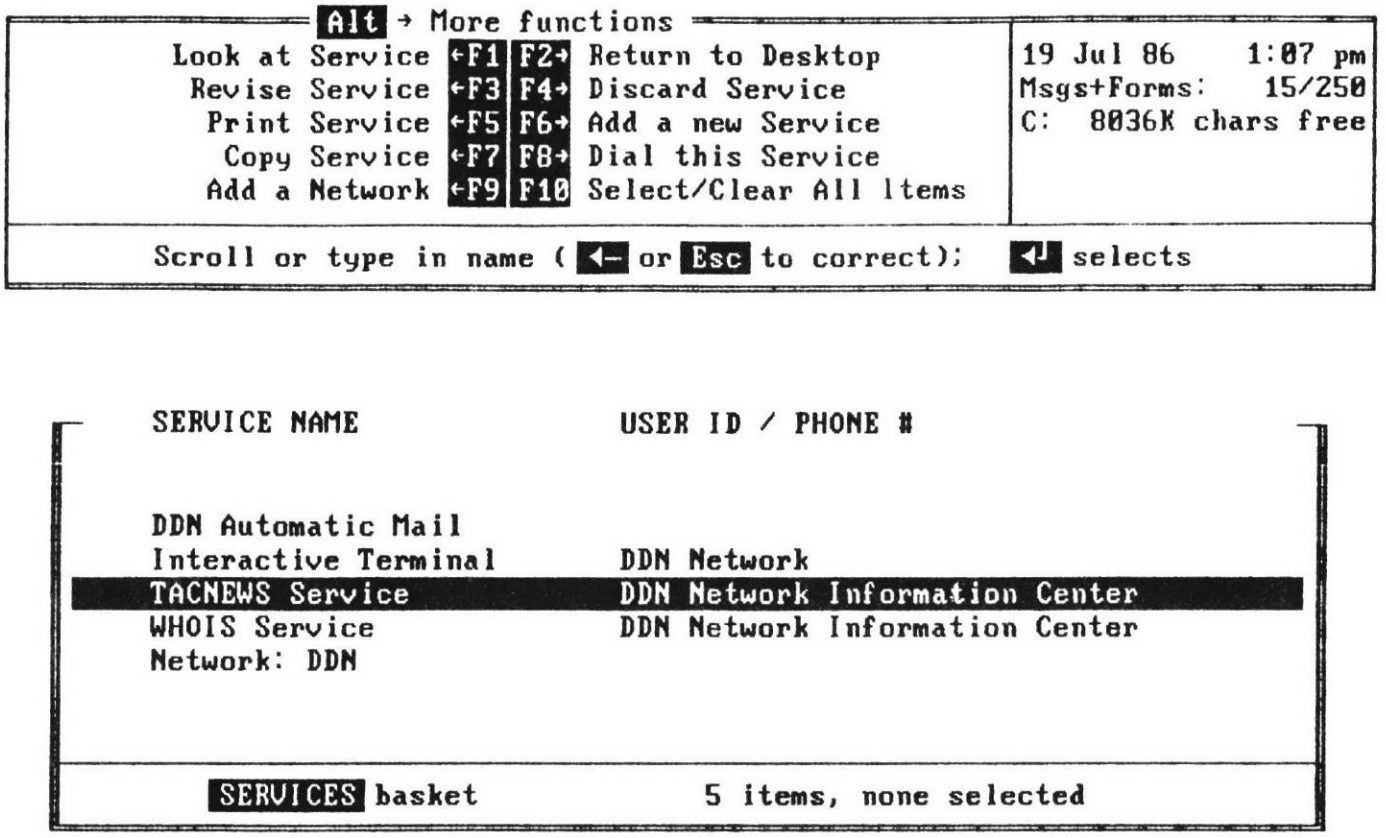


Figure 2-5: SERVICES Basket

question. Type in your DDN phone number, provided by your mailing service. If you use 9 to dial an outside number, type: **9,**

followed by the phone number. You can use dashes (e.g., 9,345-6789).

If you have several numbers available for accessing the DDN (i.e., several TAC phone numbers), these can be entered by pressing the RETURN key after each. This will provide a new line for each new number. If SAM fails to connect through the first number, it will automatically try the next, then the next, until it has tried all the numbers listed.

4. **TAC User ID.** Press the PgDn key to place your cursor in the label, "TAC User ID". Press the TAB key to move the cursor to the empty field to the right of the label. Type in your TAC User ID.
5. **Access Code.** Press the down-arrow key twice to place your cursor in the field labelled "Password 1". Type in your TAC access code. Press the down-arrow key and type it again, as indicated. For security, your access code will display as X's.
6. **Second Password.** You can skip this question. Press the down-arrow key twice and go on to the next question.
7. **Access Procedure, Keyboard Macros, Special Attributes.** While editing the service, these three groups of settings can be displayed or hidden by placing the highlighted bar on one and pressing the Return key. You will then be able to scroll downward to view or edit them.

N E T W O R K A C C E S S

Network name: DDN (Macro variable) (.sn)

Phone number:

The following fields may be used to specify parameters used for network access. They may be used from the macro language using the variable name on the right.

Figure 2-6: DDN Network Form

This form is set up to handle the most frequently used access procedure, macros[8], and special attributes. Please check that this access procedure section matches your situation (or have your system administrator check it). If it does, you can skip this question.

[8]A macro is a series of commands linked to a single function key, so that pressing that key sends the commands.

If not, revise the access procedure according to your needs. Access procedures are discussed in Section 13. For more information on Macros, see Section 14; for information on Special Attributes, see section 10.

If you make a mistake or change your mind about an answer, you can always reenter it. If you want to get rid of all the changes you have made, press the F4 key (Restore Original).

Once you are done filling out the form, press the F2 key (Return Form to Basket). SAM will save your answers. The screen will show the contents of the SERVICES basket.

2.6.3 Filling Out the "DDN Automatic Mail" Form

The next form you should fill out is the DDN Automatic Mail service form. (Service forms provide SAM with the information it needs to log onto a given service.)

Your "DDN Automatic Mail" form may differ slightly from the illustrations in this manual, depending upon which mail service you selected when you installed SAM.

If you are in the SERVICES basket, the highlighted bar will be over the "DDN Automatic Mail" line. To fill out the "DDN Automatic Mail" form,

Press the F3 key (Revise Service.) The "DDN Automatic Mail" form will appear. Fill in the form as follows:

1. **DDN Automatic Mail phone number.** If you use the DDN to access network mail (i.e., if you have already entered a phone number in the DDN Network form), you can skip this question. Press the PgDn key and go on to the next question.

   If your computer calls your mail host directly, follow these directions:

   A blinking underline cursor will appear after the first question. Type in your phone access number, if needed. For example, if you dial 9 to call an outside number, type:

   9,

   followed by the phone access number. You can also use dashes (e.g., 9,345-6789).

   If you have more than one number available, press RETURN to create a new line for each. If SAM fails to connect through the first number, it will automatically try the next, then the next, until it has tried all numbers you have listed.

2. **Press the PgDn key.** Select the Maximum Baud Rate. Most modems use either 300 or 1200 baud. Move the highlighted bar to the choice which matches your modem's speed and press the RETURN key.
3. **Press the PgDn key.** Type in your host's DDN network address. If you do not know the address and cannot obtain it locally, use the online WHOIS service to find this information or phone the DDN Network Information Center.
4. **Press the Return key.** Type in the prompt character that your host displays.
5. **Press the Return key.** Type in your host account name.
6. **Press the Return key.** Type in your host password. Press the Return key, and type in the password again, as indicated. For security, the password will display as a series of X's.
7. **Infomail Account and Password.** If you are using Infomail, type in your Infomail account name and password in the second User Name and Password fields.
8. **UNIX Mail Options.** If you are using UNIXMAIL, you can choose between creating a new .rc (options) file on your host, or using the current file. If you aren't familiar with the .rc file, leave the pointer set on "Create new .mailrc file". Otherwise, if the options file is set wrong SAM won't work.

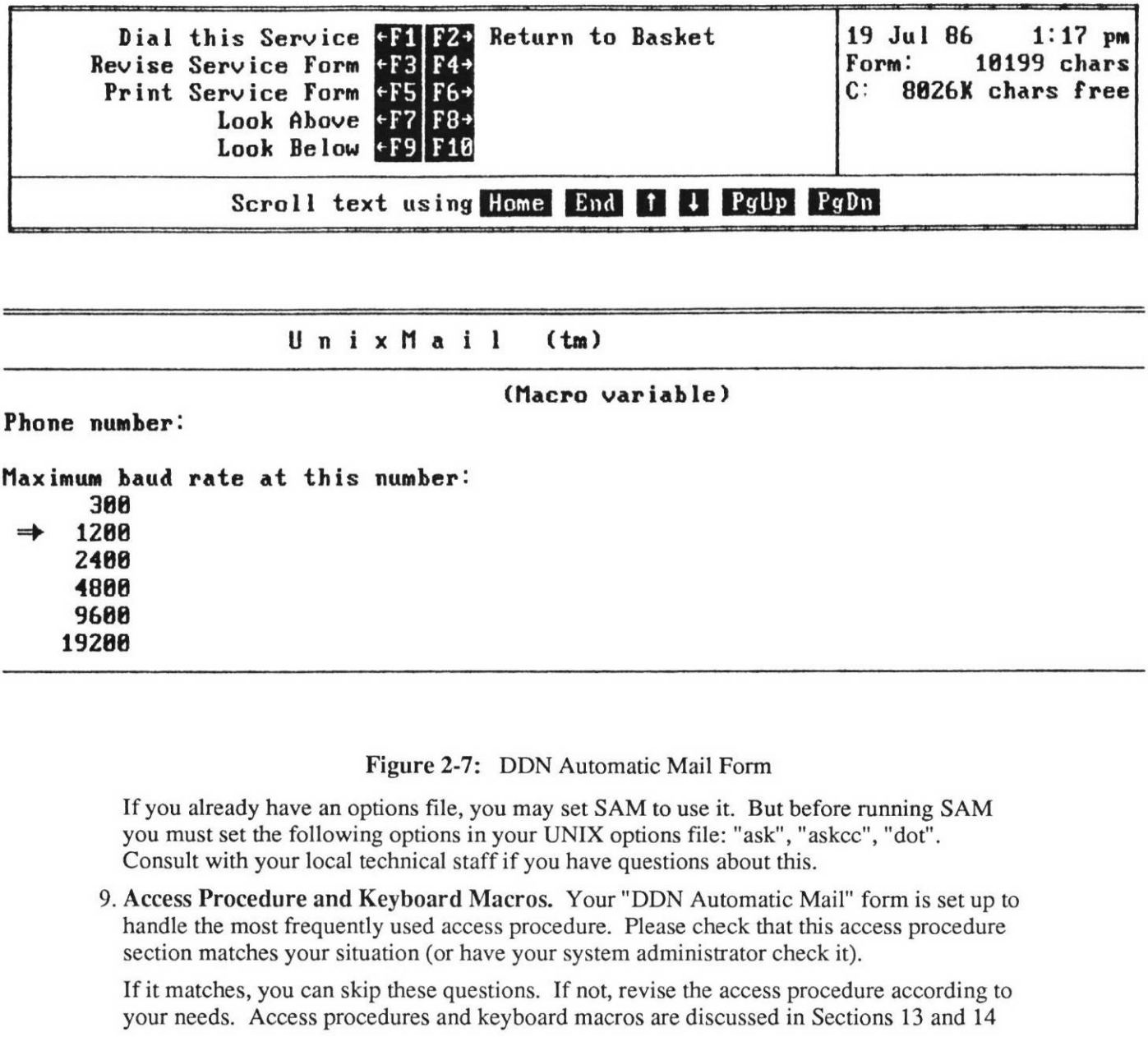


Figure 2-7: DDN Automatic Mail Form

If you already have an options file, you may set SAM to use it. But before running SAM you must set the following options in your UNIX options file: "ask", "askcc", "dot". Consult with your local technical staff if you have questions about this.

9. **Access Procedure and Keyboard Macros.** Your "DDN Automatic Mail" form is set up to handle the most frequently used access procedure. Please check that this access procedure section matches your situation (or have your system administrator check it).

   If it matches, you can skip these questions. If not, revise the access procedure according to your needs. Access procedures and keyboard macros are discussed in Sections 13 and 14

If you make a mistake or change your mind about an answer, you can always reenter it. If you want to get rid of the changes you have made, press the F4 key (Restore Original).

Once you are done filling out the form, press the F2 key (Return Form to Basket). SAM will save your answers. The screen will show the contents of the SERVICES basket.

2.6.4 Filling Out the "Interactive Terminal" Service Form

The INTERACTIVE TERMINAL service form is similar to the DDN Automatic Mail form. This is the service you will use when you want to connect to your host and work interactively, using your PC as a terminal. Follow the instructions in the previous section for completing the INTERACTIVE TERMINAL form, then refer to Section 10 for information on modifying the Special Attributes and Macros.

Note that the Access Procedure for the INTTERACTIVE TERMINAL service will need to be modified to work. The easiest way to do this is to copy the Access procedure from the DDN Automatic Mail service, once that service is working correctly.

2.6.5 Using WHOIS, TACNEWS, and BIBLIO

WHOIS, TACNEWS, and BIBLIO are useful services available to users of the DDN. They are provided by the DDN Network Information Center (NIC) under the sponsorship of the Defense Communications Agency (DCA). WHOIS serves as the network's electronic "white pages" and lets you easity find network mailboxes, host addresses, and site contacts. TACNEWS offers you a list of current network newsletters, which you can read online. BIBLIO provides a handy verification and search service for identifying and citing network documents. SAM makes the services reachable at the press of a key, and we encourage you to try them.

To use WHOIS, TACNEWS, or BIBLIO, you must first fill out the "DDN Network" form described on the previous pages. The WHOIS, TACNEWS, and BIBLIO Service forms are already completed. You do not have to modify them. To communicate with these services, move the cursor to the appropriate form and press the F8 key (Dial this Service).

Congratulations! You have now successfully installed and configured SAM. Unless you install SAM on another PC, you won't have to repeat this procedure again. SAM is now ready to use.

SECTION 3. STARTING AND LEAVING SAM

3.1 Starting SAM

Once you have installed and configured SAM, you need not do so again. For subsequent entry into SAM, use the following instructions. There are two sets of instructions: one for a PC with at least two diskette drives, and one for a PC with a hard disk drive and at least one diskette drive.

3.1.1 PC With Two Diskette Drives

1. **Insert the SAM Program Copy diskette in drive A.**
2. **Turn on your PC. If necessary, turn on your video display monitor.**
3. **Insert the Messages diskette in drive B.**
4. **After the A> prompt, type:**

   m

   and press the RETURN key.
5. **If you chose to protect SAM with a password, type it and press the RETURN key when you are prompted to do so.** For more information, see Section 2.5.2.

The Desktop display will appear.

3.1.2 PC With A Hard Disk

1. **Start your computer.**
2. **After the C> prompt, type the command name you chose during installation, and press the RETURN key.**
3. **If you chose to protect SAM with a password, type it and press the RETURN key when you are prompted to do so.** For more information, see the section describing the Preferences form in Section 2.

The Desktop display will appear.

3.2 Leaving Sam

If you want to leave SAM, follow these instructions:

1. Go to the Desktop display.
2. Press the F2 key (Exit from SAM).
3. Press the RETURN key.

Either the **A>** prompt or the **C>** prompt will appear.

SECTION 4. WRITING AND ADDRESSING MESSAGES

SAM was created to help you send and receive messages via electronic mail. This section shows you how to write and revise messages. Since each message must carry the addresses of its recipients, this section also shows you how to use the ADDRESS BOOK.

4.1 ADDRESS BOOK

The ADDRESS BOOK lists the online addresses of people to whom you often send mail. It lets you address messages quickly and accurately, and saves you from retyping them again and again. You may, but do not have to, store your recipients' addresses in the ADDRESS BOOK. See Figure 4-1.

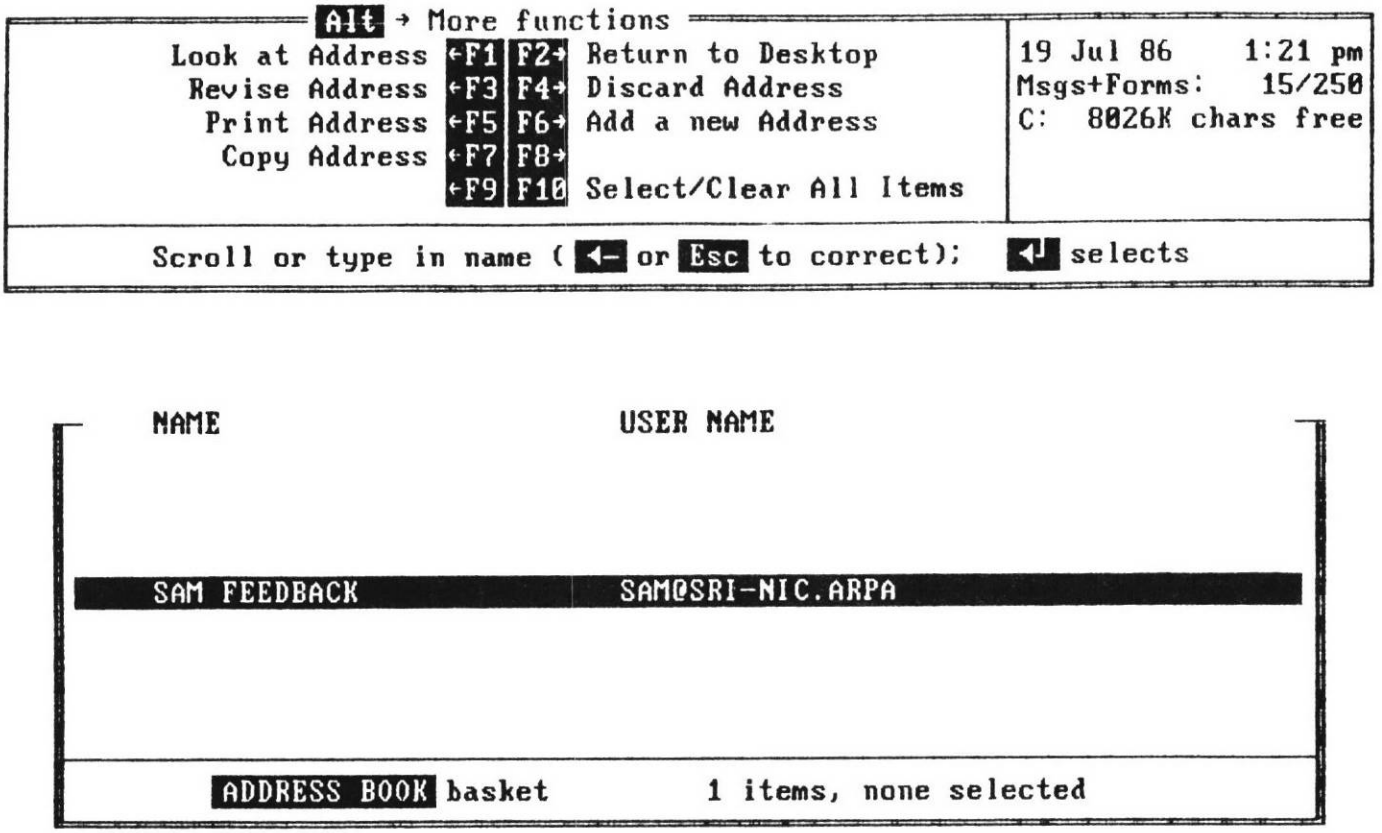


Figure 4-1: ADDRESS BOOK

Before you start, get the DDN electronic mail address for each person or organization you wish to enter in the ADDRESS BOOK. You can get these addresses from messages you recieve, by asking, or by checking with the WHOIS service. You will probably only want to store addresses that you use frequently.

To add addresses to the ADDRESS BOOK, start at the Desktop display.

1. **Move the highlighted bar to the ADDRESS BOOK.**
2. **Press the F1 key (Look at Basket.)**
3. **Press the F6 key (Add a New Address.)**

4. **Type the person or organization's name.** The ADDRESS BOOK lists names alphabetically, so you will probably want to type personal names in reverse order, with the last name first. If you prefer, you can use a keyword or nicname.
5. **Press the PgDn key.** The next question asks for the electronic mail address for the entry. Type in that person's full DDN address. This is the address that DDN Mail uses to deliver the message. (If you only know a user's name or a partial user address, you may find the full DDN address by using the NIC WHOIS service.)
6. **Press the F2 key (Save Address Entry.)**

To add more addresses, return to Step 3.

If you make a mistake or change your mind about an item, you can always fix it. Use the arrow keys and the Home, End, PgUp and PgDn keys to move the highlighted bar or blinking cursor to the mistake. Then enter the correct answer. If you want to get rid of all of the changes, press the F4 key (Restore Original).

To revise an address, you should be in the ADDRESS BOOK. Then, follow these instructions:

1. Move the highlighted bar to the address you want to change.
2. Press the F3 key (Revise Address). You can change any or all answers on the form.
3. Press the F2 key (Save Address Entry) to save the revised address.

4.2 Writing Messages

To create a new message, return to the Desktop display, if you have not already done so. Then, follow these instructions:

1. **Move the highlighted bar to the OUT basket.**
2. **Press the F1 key (Look at Basket.)**
3. **Press the F6 key (Add a New Message.)**
4. **Type the subject of your message.** The first 25 characters of the subject will be displayed in the OUT basket.
5. **Press the PgDn key. The cursor will move to the "To:" field.**
6. **Type in the first recipient's online address.**

   To send the message to more than one person, press RETURN after you enter the first recipient's address. A second line will appear. Type the second recipient's address on the second line. Add each additional recipient's online address on a new line.

   You should list frequently-used addresses in your ADDRESS BOOK. You can place these recipients in your "To:" field automatically by following steps 7 through 9, below.
7. **With the cursor on the "To:" line, press the F3 key (Look up Names.)** The ADDRESS BOOK will appear. See Figure 4-2.
8. **Move the highlighted bar to the first desired recipient.** Press the RETURN key. An arrow will appear next to your choice. If you change your mind, press the RETURN key again. The arrow will disappear.
9. **Move the highlighted bar to the next recipient, and press the RETURN key.** You can choose as many recipients as you like. Note that you must mark (with the RETURN key) the addresses you want transferred to the "To:" field; the address that is highlighted will not be moved unless marked.

   You can locate a name in the ADDRESS BOOK by typing the first few letters of the name.

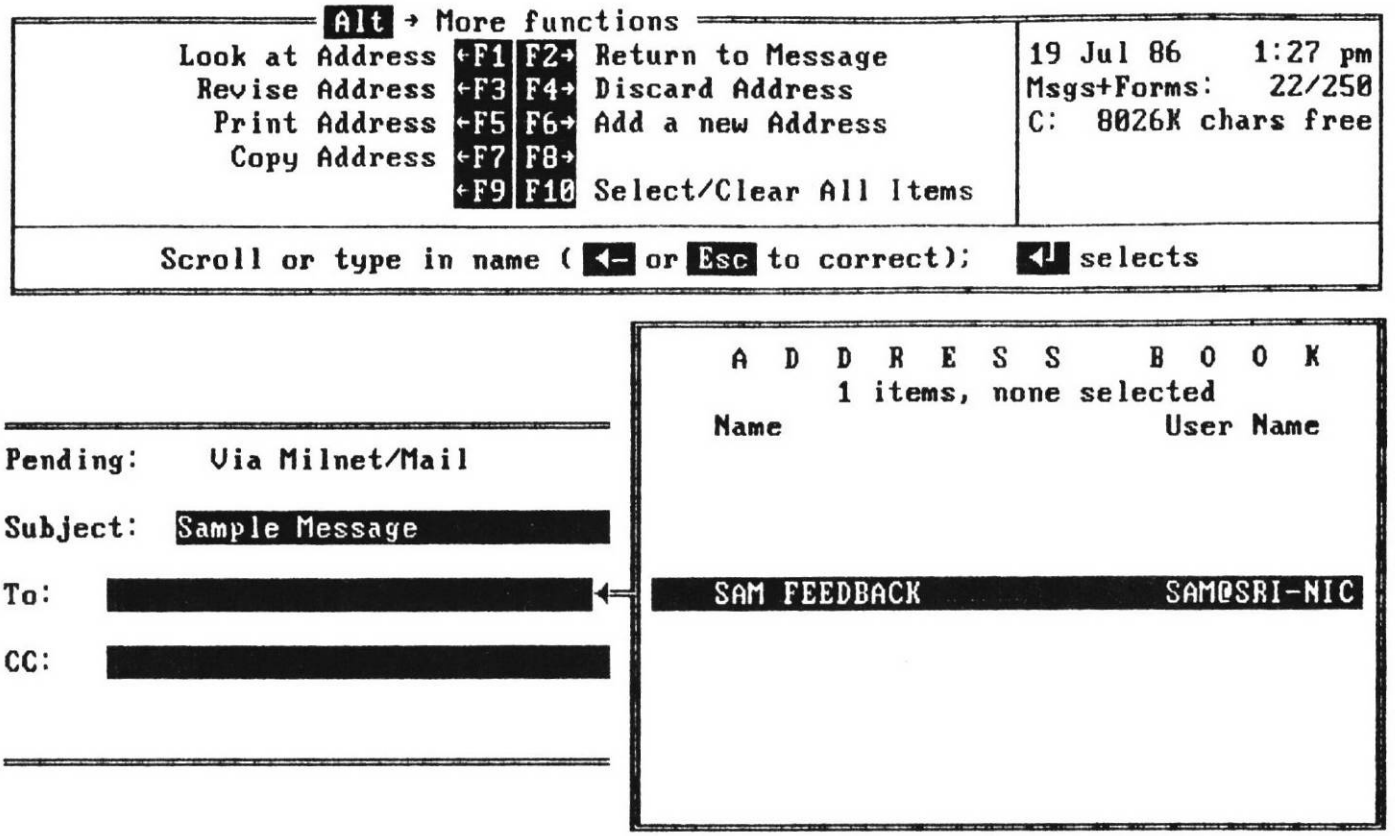


Figure 4-2: Look Up Names

10. **Press the F2 key (Return to Message.)**

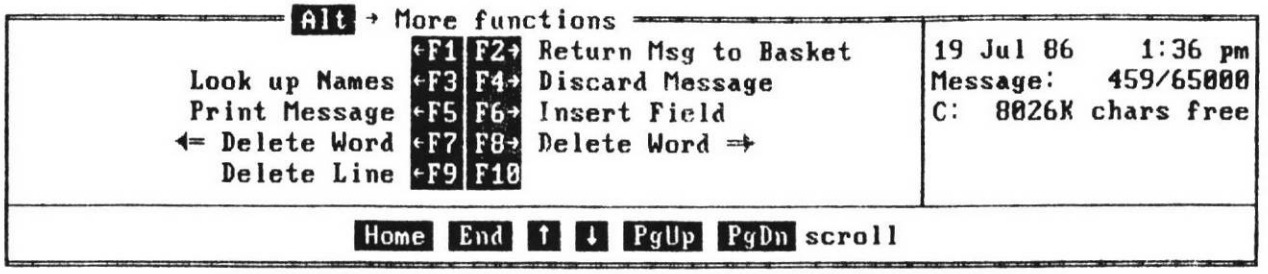
11. **Press the PgDn key or press the RETURN key twice.** The cursor will move to the "CC:" field. If you want someone to receive a courtesy copy of this message, repeat steps 6 through 9.

12. **Press the PgDn key twice.** You are now ready to type in your message.

    Type as you would on a typewriter, with this exception: press the RETURN key only to start a new paragraph. SAM will end each line within a paragraph automatically. Correct mistakes by using the editing keys described later in this section.

13. **Press the F2 Key (Return Msg to Basket.)** You may now send your message, if you wish (see Section 5).

NOTE: SAM's editor cannot create messages containing more than 64,000 characters (about 30 pages). Messages larger than this should probably not be sent via electronic mail, but instead moved by file transfer. This is because some DDN Mail programs have a size limit on messages they will accept. However, SAM can send large messages if they are created as a text file, then brought into SAM using the <Alt>F7 key (Copy a File into Message). Figure 4-3, shows a sample message.




```
Pending:    Via Milnet/Mail                          19Jul86  1:32pm

Subject:  Meeting Tomorrow

To:       Sherman@sri-nic.arpa

CC:

Gary,

Let's plan to discuss the problems that have shown up this weekend
during the meeting tomorrow.

Gwenn
```

Figure 4-3: Sample Message

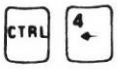
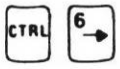
4.3 Revising Messages

If you want to revise a message, you should be in the basket containing that message. Then, follow these instructions:

1. **Move the highlighted bar to the message.**
2. **Press the F3 key (Revise Message.)** You may change any of message's information. When you are entering or revising the text portion of a message, many of the keys on the keyboard have special functions. Refer to the following list.

NOTE: Use Revise Message to fill in the SAM registration form in the REGISTRATION basket.

Cursor Movement

Key	Moves the Cursor
	Up one line
	Down one line
	Left one character
	Right one character
	Left one word
	Right one word
	To the next tab stop
	To the start of the next line
	To the end of the text
	Up 8 lines
	Down 8 lines

Deletion

Key	Deletes
F10	Character to the left of cursor
F4	Character at cursor position
F5	Word to the left of cursor
F6	Word to the right of cursor
←	Line that contains cursor

Other Functions

Key	Function
Del	Restores last line erased using the F9 key
F7	Restores original message; discards all revisions
F8	Prints message (see Section 6)
F9	Inserts a RETURN to the left of cursor

In addition to the editing functions listed above, you can use two other functions while you revise a message--Copy Message to File and Copy File into Message.

Key	Function
ALT F7	Copies the body of the message into a file you specify. The file is stored in your Work on Files directory (if you have a hard disk) or in the diskette in drive A (if your PC has two diskette drives).
ALT F9	Copies a file you select from your Work on Files directory into the body of the message at the cursor position. Note: If you copy a word-processed file into your message, SAM will strip out the control codes and leave only ASCII text.

SECTION 5. USING SAM SERVICES

SAM comes with five services already installed. These are "Automatic Mail", "Interactive Terminal", "TACNEWS", "WHOIS" and "BIBLIO". This section assumes that you have filled out all of the forms discussed in Section 2.

5.1 Sending and Receiving Messages

If you ask SAM to send your messages, it will send all the messages in the OUT basket, whether marked or not. Make sure that the OUT basket contains all of the messages you want to send, and only the messages you want to send. (This is a good time to practice by becoming a registered SAM user. Just fill out the form contained in the REGISTRATION basket, then move it into the OUT basket.)

If your modem is not built into your PC, make sure that the modem is turned on and connected to the PC.

Then, follow these instructions:

1. **Go to the Desktop display, if you are not already there.**
2. **Press the F6 key (Send and Receive Mail.)** The "Mail Delivery Options" form will appear. See Figure 5-1.

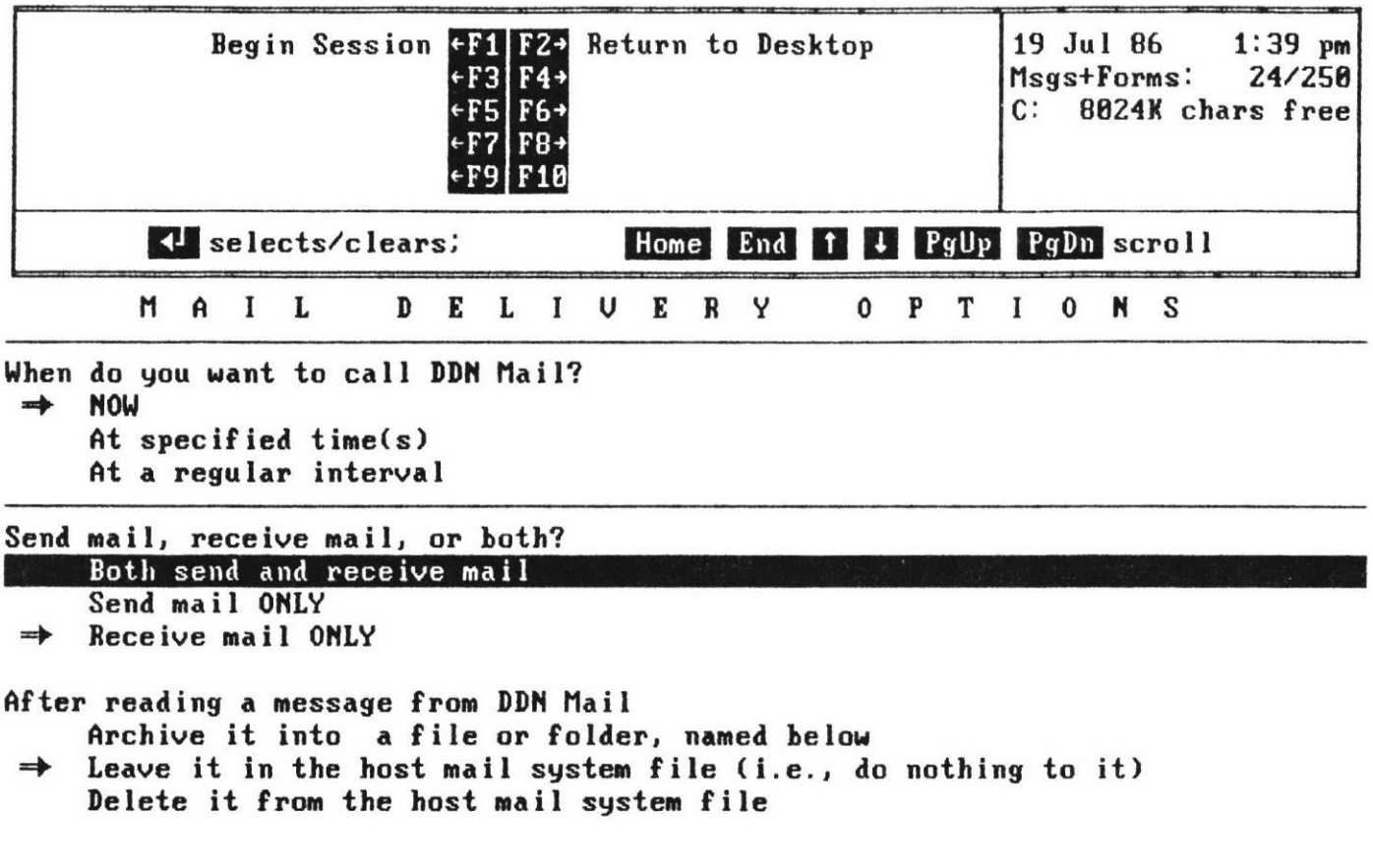


Figure 5-1: Mail Delivery Options Form

3. **Tell SAM that you want to call network mail now.** Select the answer "NOW" to the first question.

4. **Press the PgDn key.** The next question is, "Send mail, receive mail, or both?" Move the highlighted bar to your choice and press the RETURN key. The usual choice is "Both send and receive mail".
5. **Press the PgDn key.** The next question asks whether to archive (save) the message in a host file, delete the message, or leave it in your mail program. Move the highlighted bar to your choice and press the RETURN key.
6. **Press the PgDn key.** The next question is, "Record log in messages into the IN basket?" Move the highlighted bar to your choice and press the RETURN key.
7. **Press the PgDn key.** The next question is, "What if a call does not go through?" First, type in the number of times that the system should retry the call. Then press the PgDn key, and type in the number of minutes the system should wait between retries. (Note that the shortest time SAM will wait to redial is 3 minutes.)
8. **Press the F1 key (Begin Session.)**

SAM will tell your modem to call the DDN. When the connection to your mail host has been made:

- If you asked SAM to send messages, it will send the messages in your OUT basket to DDN mail and put a copy of each message into your SENT basket.
- If you asked SAM to receive messages, it will receive your incoming mail from DDN mail and put it in the IN basket.

When it has finished, SAM will tell your modem to hang up the phone. The Desktop display will reappear.

NOTE: Sometimes SAM will be unable to connect to your host, due to noise on the phone line or for other reasons. When this happens SAM will automatically "time out" after a few minutes, hang up, and try again. There is no need for you to interrupt the procedure, although if you are in a hurry you may do so by pressing ESC and then starting over again.

5.2 "Interactive Terminal"

There will be times when you want to deal with your mail manually, or want to log on to your host and work with programs other than mail. The Interactive Mail service will automatically log you in to your host, then will allow you to work on the host just as you would it you were using a terminal. For your convenience, this service also emulates a VT100 terminal. (If you wish to adjust any of the terminal parameters, see the VT100 Settings section at the bottom of the form.)

To use this service:

1. **Go to the Desktop display, if you are not already there.**
2. **Press the F8 key (Dial a Service).** You will drop into the Services basket.
3. **Position the highlighted bar on the "Interactive Terminal" service.**
4. **Press the F8 key (Dial This Service).** SAM will go into terminal mode and begin dialing. You can watch the macro commands being issued and responses to them at the top of the screen.
5. **Once the access procedure has finished and SAM has successfully logged in to the host, you will be able to work on the host. You can see the SAM function keys chart at any time by pressing the Ctrl (control) and Home keys simultaneously.**
6. **If the login fails, press ESC to abort the access procedure, then start again.**
7. **When you are finished working, logout of your host and then press the F2 key (hang up).**

5.3 TACNEWS, WHOIS, and BIBLIO Services

TACNEWS, WHOIS, and BIBLIO are two services the NIC provides the DDN community. These services are generally available on you host. SAM lets you use these services directly on the NIC without logging in to your host.

TACNEWS provides general information of interest to TAC users, such as the current list of TAC phone numbers. **WHOIS** accesses the current database of registered DDN users, and is useful for finding DDN mailboxes for people and network addresses for hosts. **BIBLIO** accesses a database containing citations for information on DDN (and other) protocols.

All three programs have built-in help available once you are connected to them. For further assistance, contact your Host Technical Liaison, or call the DDN Network Information Center.

To use one of these services:

1. **Go to the Desktop display, if you are not already there.**
2. **Press the F8 key (Dial a Service).** You will drop into the Services basket.
3. **Position the highlighted bar on the service you wish to access.**
4. **Press the F8 key (Dial This Service).** SAM will go into terminal mode and begin dialing. You can watch the macro commands being issued and responses to them at the top of the screen.
5. **Once the access procedure has finished and SAM has successfully logged in to the host, you will be able to use the service. You can see the SAM function keys chart at any time by pressing the Ctrl (control) and Home keys simultaneously.**
6. **If the login fails, press ESC to abort the access procedure, then start again.**
7. **When you are finished working, press the F2 key (hang up).**

SECTION 6. READING, PRINTING AND ANSWERING MESSAGES

When SAM receives your incoming messages, it puts them in the IN basket. You can then read, print and answer those messages. See 6-1.

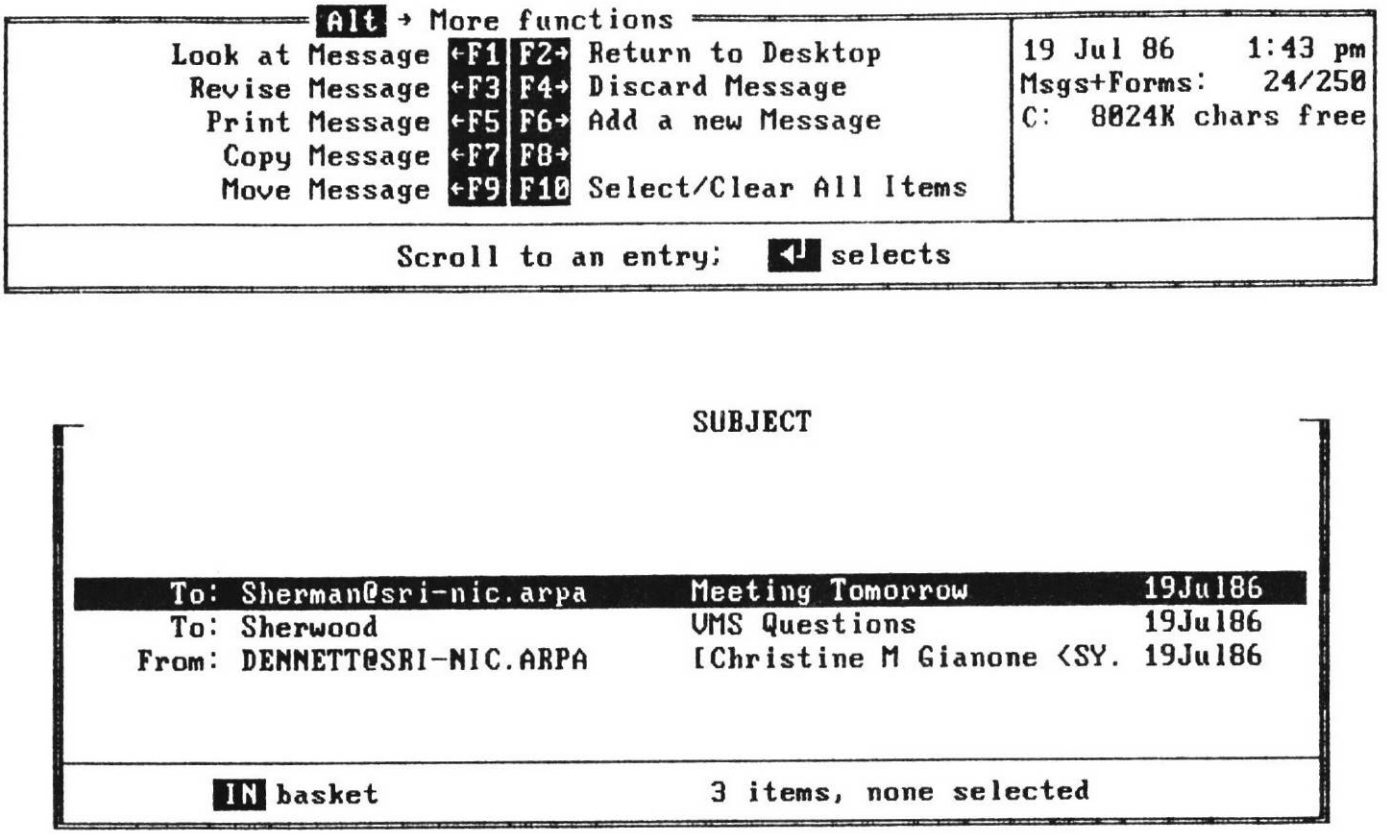


Figure 6-1: In Basket

6.1 Reading And Printing Messages

If you want to print a message, make sure that your printer is connected to your PC and turned on. Also, you must have already filled out the forms shown in Section 2.

To read or print a message, follow these instructions:

1. **Return to the the Desktop display, if you have not already done so.**
2. **Move the highlighted bar to the IN basket.**
3. **Press the F1 key (Look at Basket).** The IN basket will appear.
4. **Move the highlighted bar to the message you want to read or print.**
5. **Press the F1 key (Look at Message).** Use the keys on the numeric keypad to see more of the message.

   NOTE: If a message is too large for SAM, it will be written to a DOS file. In this case, when you look at the message you will see "Message too big, received as

MSG####1.TXT". To view the message, return to the desktop and select F9 "Work on Files". You will be shown a display of files, from which you can select the one containing your message. View it by pressing F1 ("Look at file").

6. **To print the message, press the F5 key (Print Message).** Make sure that your printer is ready to print, and then press the RETURN key.

   To stop the printing, press the Esc key.

7. **To read the next or previous message, press the F9 key (Look Below) or the F7 key (Look Above).**
8. **To return to the basket, press the F2 key (Return to Basket).**

You can also print messages without reading them:

1. **Return to the IN basket, if you have not already done so.**
2. **Move the highlighted bar to the message you want to print.** If you want to print all your messages, press the F10 key (Select/Clear All Items).
3. **Press the F5 key (Print Message), and then press the RETURN key.** Your message(s) will be printed.

6.2 Answering Messages

You can reply to the author of a message. To do so, follow these instructions:

1. **Return to the IN basket, if you have not already done so.**
2. **Move the highlighted bar to the message to which you want to reply.**
3. **Press the F8 key (Reply to Author).**

SAM moves over to the OUT basket and creates a new message addressed to the author. Complete the message as you would any other.

When you are finished, press the F2 key (Return to Basket) to save the message. Note that you are now in the OUT basket, and that your reply is ready to be sent.

6.3 Forwarding Messages

You can forward to someone else the text of messages you receive. To do so, follow these instructions:

1. **Return to the IN basket, if you have not already done so.**
2. **Move the highlighted bar to the message which you want to forward.**
3. **Press the <Alt>F10 key (Forward Message).**

SAM moves over to the OUT basket and creates a new message with the text of the old message already inserted. Type the recipient's address into the "To:" field, then add any comments you wish above the forwarded message.

When you are finished, press the F2 key (Return to Basket) to save the message. Note that you are now in the OUT basket, and that your reply is ready to be sent.

SECTION 7. MANAGING MESSAGES

You can tell SAM to move or copy a message. When SAM moves a message, it takes the message out of its current basket and puts it in another basket. When SAM copies a message, it makes a duplicate of the message, puts the duplicate in another basket, and leaves the original message where it was.

To move or copy a message, follow these instructions:

1. **Return to the Desktop display, if you have not already done so.**
2. **Move the highlighted bar to the basket that holds the message you want to move or copy.** If you are not sure which basket holds the message, you can press the F7 key (Print Desktop Summary) to print out a summary of the contents of all of the baskets. (You must have a printer attached to your PC to use this function.)
3. **Press the F1 key (Look at Basket).**
4. **Move the highlighted bar to the message you want to move or copy.**
5. **If you want to move the message, press the F9 key (Move Message).**

If you want to copy the message, press the F7 key (Copy Message).

The Desktop display will appear, along with a pointer. Use the arrow keys to move the pointer to the destination basket. Then press the RETURN key. The message will be moved or copied to the destination basket.

You can also move or copy a group of messages, as long as they are all coming from a single basket and going to a single basket. To do so, replace Step 4 with the following:

4. **Move the highlighted bar to the first message you want to move or copy.** Press the RETURN key to mark the message. Then move the highlighted bar to the next message you want to move or copy, and press the RETURN key again. Repeat this step for every message you want to move or copy.

   NOTE: After a message is copied or moved, SAM will normally return to the desktop. If you wish, you can set SAM to instead return either to the basket the message came from, or to remain in the basket the message went to. See Section 2.5.2 for details.

7.1 Filing Messages

The upper 8 baskets in the Desktop display are used by SAM for its operation. The lower 16 baskets are designed to hold messages, and can be used as your personal filing system. You can move or copy messages between any of the lower 16 baskets and any basket in the Desktop display.

For example, you could use one basket to hold all form letters, another to hold personal messages, a third to hold all correspondence with a business associate, and so on.

Each of the upper 8 baskets bears a label such as "IN" or "ADDRESS BOOK". You can choose your own label for each of the lower 16 baskets. To label one of these baskets, follow these instructions:

1. **Return to the Desktop display, if you have not already done so.**
2. **Move the highlighted bar to the basket you want to label.**
3. **Press the F3 key (Label Basket).**
4. **Type the label, and then press the RETURN key.** The basket is now the labelled.

NOTE: Once you have sent in your SAM registration form, you can re-label the REGISTRATION basket for your own use.

7.2 Discarding And Retrieving Messages

After you have used SAM for a while, you may find your baskets full of outdated, unneeded messages. Since those messages are taking up valuable disk space, you should discard them.

To discard messages:

1. **Return to the Desktop display, if you have not already done so.**
2. **Move the highlighted bar to the basket holding the message you want to discard.**
3. **Press the F1 key (Look at Basket).**
4. **Move the highlighted bar to the message you want to discard.** If you only want to discard one message, continue to the next step.

   To discard more than one message, mark each by moving the highlighted bar to it and pressing the RETURN key. To discard ALL the messages in the basket, press the F10 key (Select/Clear All Items).
5. **Press the F4 key (Discard Message).** Then press the RETURN key. SAM will move the message(s) to the WASTE basket.

If you discard a message and realize later that you still need it, you can retrieve it, unharmed, from the WASTE basket[9] Follow these instructions:

1. **Return to the Desktop display, if you have not already done so.**
2. **Move the highlighted bar to the WASTE basket.**
3. **Press the F1 key (Look at Basket).**
4. **Move the highlighted bar to the desired message.** You can retrieve more than one message at a time, since you are using the Move function. For more information, see the section on moving and copying messages at the beginning of this chapter.
5. **Press the F9 key (Move Item).** Move the message(s) to another basket, where it will be safe.

7.3 Shredding Messages

When you discard an item, it is not removed from your computer; it is simply moved to the WASTE basket. The WASTE basket holds discarded messages, forms and addresses, as well as the original copies of revised messages, forms and addresses (see Figure 7-1). To permanently remove any or all of these items from your disk you must shred them.

To shred an item, follow these instructions:

1. **Open the WASTE basket.**
2. **Move the highlighted bar to the item to be shredded.** To check the contents of the item, press the F1 key (Look at Item). To shred all items, press the F10 key (Select/Clear All Items).
3. **Press the F4 key (Shred Item).** This is your last chance to change your mind. When you press the RETURN key, the item(s) will disappear forever.

You can also tell SAM to shred items automatically, by changing the answer to the second question on the Preferences form in the YOUR PC basket.

---

[9]Unless you have "shredded" it, or have your WASTE basket set to automatically discard the oldest messages.

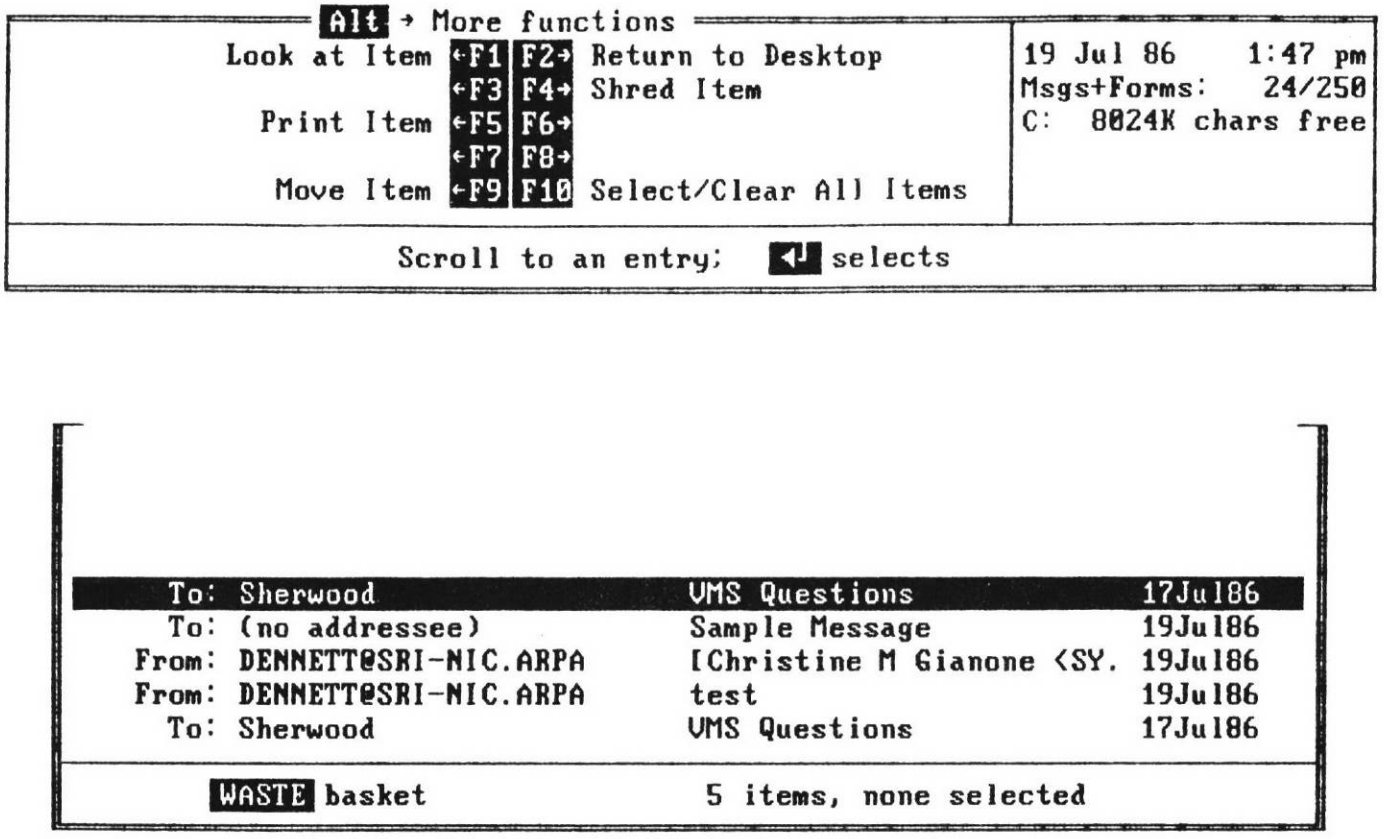


Figure 7-1: Waste Basket

7.4 Archiving Messages

SAM can hold up to 95 messages when installed on floppy diskettes, or up to 250 messages on a hard disk (if you have enough space on the disk). Once you reach these limits, you will need to either delete or archive the messages. Archiving may be done by copying messages into text files. To do this:

1. Go into the basket containing messages you wish to archive.
2. Use the RETURN key to mark the first message you want to archive.
3. Press Alt-F7 (Copy Message to File). You will be asked what you want to name the file. Supply a name and press RETURN.
4. Press F4 (Discard Message) to move the message into SAM's WASTE basket. When you have finished archiving messages, go into the WASTE basket and shred the messages contained there to free up space for new messages.

SECTION 8. MAIL SECURITY

Normal electronic mail security guidelines, such as not sending classified information through e-mail, apply to SAM as much as to a dumb terminal. However, SAM's extra capabilites require you to exercise some caution beyond these.

To work unattended, SAM contains the account names and passwords for your TAC and mail accounts. This gives SAM the ability to access your accounts, no matter who is running the program. For this reason, it is very important, once you have SAM up and running, to turn on the SAM password. When this password is enabled, it will be required to start SAM and to run any of the services.

To protect passwords from casual snoopers, they don't display in the service forms, and are stored in encrypted files on your disks. However, if you feel uncomfortable having your passwords stored by SAM, you can instead use the "prompt hide" macro (Section 13.5.2) command to make SAM stop and ask for your password each time a service is run.

Additionally, if you are using a floppy disk-based version of SAM, you can remove your floppies and lock them up when not running SAM.

Software is available to encrypt files, both for storage on your disk or for secure transmission. For more information on these programs and protecting your data, consult with your site Security Officer.

SECTION 9. COMMUNICATING WITH ELECTRONIC MAIL HOSTS

In the previous sections, the only type of communication discussed was the automatic transfer of messages, where SAM sends all of the messages in the OUT basket to DDN mail and receives all of your messages from DDN mail to your IN basket. In this section, we will show you how to log in and perform other types of communication with your mail host computer.

9.1 Dialing And Logging In

To communicate with your mail host, begin by calling it and logging in. Follow these instructions:

1. **Go to the Desktop display, if you are not already there, and press the F8 key (Dial a Service).** The SERVICES basket will appear.
2. **Move the highlighted bar to "Interactive Terminal". Then press the F8 key (Dial this Service).** SAM will use your computer and modem to call your mail host. When the host answers the phone, SAM will give it your user name and password. After a few moments, your DDN Mail program prompt will appear. (See Figure 9-1).

You are now logged in to your DDN host.

```
CtrlHome→Menu | Online | 0:04:28 | Print off | Buffer:    285 chars | C: 8024K free | NUM
@i 4
o /73

TAC Userid: SCD-SRINIC
Access Code:

Login OK
TCP Trying...Open
+-------------------------------------------------------------------------+
|  For TACNEWS, type:                          TACNEWS <carriage return>  |
|  For user and host information, type:        WHOIS HELP <carriage return>|
|  For NIC information, type:                  NIC <carriage return>      |
|  Report system problems to ACTION@SRI-NIC or call (415) 859-5921        |
+-------------------------------------------------------------------------+

 SRI-NIC, TOPS-20 Monitor 6.1(6405)-4
 The system will go down Sat 19-Jul-86  4:00pm until Sat 19-Jul-86  4:05pm
 for Free UDP queues
@dennett
Password:
 Job 13 on TTY151 19-Jul-86  1:51pm
 Previous LOGIN: 19-Jul-86  1:47pm
 End of <DENNETT>LOGIN.CMD.136
@
```

Figure 9-1: Logging in to Your Host

9.2 Commands

Once you have logged in, you may enter commands, or use SAM to give certain commands for you.

To see the Function Keys window, which shows SAM's commands, hold down the Ctrl key and press the Home key. You can create your own custom function keys using those that are unassigned or which have functions you don't need (see section 13). Here is a brief description of the function keys' commands:

F1: Keyboard Echo On/Off

Keyboard Echo tells your computer to display on screen the letters you type. Some services automatically display the letters you type; others do not. If you are connected to another computer and can't see the letters you are typing, try turning Keyboard Echo on. If each letter you type is doubled, turn Keyboard Echo off.

F2: Hang Up Disconnects SAM from the service and returns you to the SERVICES basket. If you have an autodial modem, SAM tells it to hang up. (This does not log you out from your host.)

F3: Print On/Off Turns the printing of what appears on the screen on and off. To start printing, press F5. To stop printing, press F5 again.

F4: Capture Buffer On/Off

The buffer is a section of your PC's disk storage that SAM uses to hold a copy of everything that appears on the screen. Each service starts with the buffer turned on. This key toggles the buffer between on and off. Note that turning the buffer off will increase display speed.

F5: KERMIT Send 8-bit File

KERMIT can be used to upload both text and binary (8-bit) files. However, when uploading binary files, an extra command must be sent to the host KERMIT. This key starts the host KERMIT in binary mode, then starts the DOS KERMIT and sends the file. You will be asked to supply the name of the file you wish to transfer.

F6: Clear Buffer This key clears the contents of the capture buffer.

F7: Send File with KERMIT

This function lets you send a text file from your PC to the host computer using KERMIT error checking. KERMIT will only work if your host also has a version of KERMIT available (your Host Technical Liaison can tell you if it does).

F8: Save Buffer Copies the contents of the Buffer (see F6) into a permanent file on your Work on Files diskette or directory. After you press F8, type in a file name for the new file.

F9:Receive File w/ KERMIT

This function lets you copy a file from your host to your PC, using KERMIT error checking. Whenyou press it, you swill be asked for the name of the file you wish to copy. KERMIT must be available on your host computer to use this function.

F10: Send File - no verify

Sends a file to the service. When you press F10, the screen displays a directory of your Work on Files disk. To send a file, scroll to the desired file and press the F10 (Send File) function key.

Alt-F7: Send file w/ XMODEM

This function allows you to send files to the service using XMODEM error checking. When you press F9, the screen displays a directory of your Work on Files disk. To send a file, scroll to the desired file and press the Send File function key.

The XMODEM function works only when you communicate with a service that has

standard XMODEM protocol. (Find out from the Host Technical Liaison whether you can use XMODEM.)

Alt-F9: Receive file w/ XMODEM

This function allows you to receive files from the host compuer using XMODEM error checking. To begin receiving a file, type in a file name.

The XMODEM function works only when you communicate with a service that has standard XMODEM protocol. (Find out from the Host Technical Liaison whether you can use XMODEM.)

Alt-F10: Execute DOS Command

Push to the DOS prompt without logging out or breaking the connection. This command will only work if you have enought free memory.

CONTROL-F3: Revise Service Form

Edit the service form to change variables such as pacing, VT100 emulation, etc. Changes made take effect as soon as you save the form.

CONTROL-F6: Revise a Macro

Record a new macro for any function key. Note that if you choose a funtion key that already has a macro, the old macro will be lost.

SECTION 10. COMMUNICATING WITH OTHER COMPUTERS

This section will show you how to communicate with nearly any other computer.

10.1 Adding A Service

To communicate online with an information service or any other computer system you need a Service form for that system. Each Service form holds the information needed to log in to one system. (See Figure 10-1.)

NOTE: If you want to communicate with a host on the DDN other than your mail host, the easiest way to do so is as follows:

1. **Use the F7 key to copy the "Interactive Terminal" form.** There will now be two service forms with the same name.
2. **Revise one of the forms.** Change the name of the form and the host address, prompt, account name and passwords.
3. **The new form is now ready to use.**

The following example will show you how to fill out a new Service form. In the example, the service we want to communicate with is THE SOURCE (a commercial mail and database service).

Before continuing, you'll need the following information:

- The service's phone number,
- The service's baud rate (rate of communication; either 300 or 1200 in most cases),
- Your user ID, password, and login procedure for the service.

Then, follow these instructions:

1. **Go to the the Desktop display,** if you have not already done so. Then press the F8 key (Dial a Service). The SERVICES basket will appear. (You can also do this by moving the cursor to the SERVICES basket and pressing the F1 key.)
2. **Press the F6 key (Add New Service).** The screen displays an empty Service form.
3. **In the "Name of Service" line, type the name of your service.**

   In our example, we would type:

   THE SOURCE

4. **Press the PgDn key. Type in the phone number of the service.**

   In our example, we would type our local SOURCE access phone number. SAM will ignore hyphens, spaces, and parentheses.

5. **Press the PgDn key. Select the baud rate your service uses.** Press the RETURN key. An arrow indicates the selected baud rate.
6. **If your service requires that you emulate a VT-100 terminal, press the PgDn key and select that option.** See the end of this section for more information on VT-100 emulation.
7. **Continue to the following section, "Dialing and Logging In".**

10.2 Dialing And Logging In

You should still be editing your Service form. Now you are going to dial the service, log in, and record your login for future use.

In our example, you need your SOURCE user ID and password. If you use another service, you need the login information that service requires.

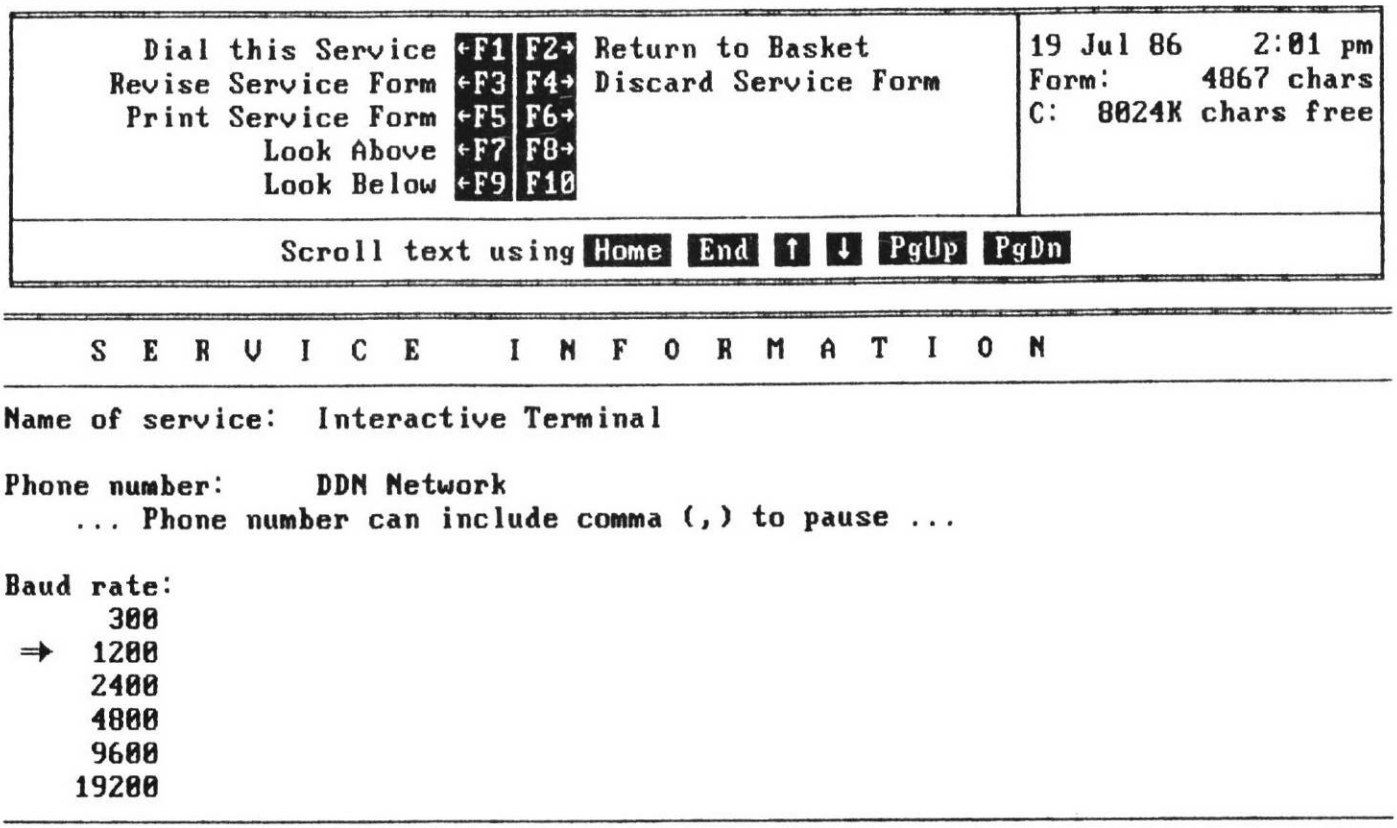


Figure 10-1: Service Form

Follow these instructions:

1. **Press the F1 key (Dial this Service), and then type:**

   L

   Do not press the RETURN key. SAM will dial the service.

   If you get a busy signal or a "No Answer" message, press the ESC key. Then, try again by pressing the F1 key (Dial this Service).

   If you get a "NO CARRIER" message, check the phone number you entered in the Service form. If the number is correct, check the baud rates you entered in your PC Configuration and Service forms. Make sure the phone jack is connected to your computer and phone line. Then try again.

   If you still can't get through, make sure the service you're calling is operating. Check your modem for hardware problems.

2. **When the service answers, log in.** If you get into trouble during your login, press the F2 key (Hang Up). Then press the RETURN key to go back to the Service form. Begin again with Step 1.

   When THE SOURCE answers, you will see:

   Connected to THE SOURCE
   >

Type the letters "ID" followed by a space, then your user ID followed by a space, then THE SOURCE password, and then press the RETURN key. For example:

ID AAA123 MYPASS

THE SOURCE displays its welcome message and main menu. You have now logged into THE SOURCE.

3. **When you finish logging in to your service, hold down the Ctrl key and press the End key.** This tells SAM to stop recording your login.

The next time you use the F1 key to dial this service, press the RETURN key instead of the letter "L". SAM will log you in automatically. If the automatic login is unsuccessful, see Section 13 (Auto Logins and Keyboard Macros).

10.3 Retrieving Information

This section shows how to record information from a service and store it in a file in your Work on Files diskette or directory.

You should now be communicating with your service. In our example, THE SOURCE main menu would be on your screen.

At any time while communicating, you can hold down the Ctrl key and press the Home key. The Function Keys window will appear at the top of the screen, showing SAM's online functions. Pressing ESC will resume your online session.

Follow these instructions:

1. **Give your service all but the final command needed to display the information you want to retrieve.**

   In our example, THE SOURCE prompts:

   Enter item number or HELP

   Type:

   6

   and press the RETURN key. THE SOURCE prompts:

   >

   Type:

   AIRSCHED-D

   and press the RETURN key. THE SOURCE prompts:

   Domestic Air Schedules

   While you communicate online, SAM puts a copy of whatever is on the screen into a section of your PC's disk storage called the Buffer. In the following steps you will remove all the data stored in the Buffer so far. Then you will retrieve the desired information from your service, and save it in a file on your Work on Files diskette or directory.

2. **Press the F6 key (Clear Buffer). Press the RETURN key.** SAM clears the buffer and puts you back to the Online screen again.

3. **Give your service the final command needed to display the information you want to retrieve.**

   In our example, THE SOURCE prompts you to enter cities and times of departure. Type departure city / destination city and time. For example:

SAN FRANCISCO,CA/NEW

YORK,NY 0900 AM-1130 AM

Press the RETURN key and your flights will be listed. After displaying the flights, THE SOURCE prompts for another air schedule. Type:

Q

and press RETURN. The screen displays the > prompt.

4. **Press F8 (Save Buffer).** This copies the contents of the Buffer into a permanent file on your Work on Files diskette/directory. At the moment, the Buffer contains the information that you retrieved from your service.

   The Help window prompts:

   Save buffer, file name:

   Type in a name for the new DOS file and press the RETURN key. DOS file names can have up to 8 characters, then a period (.), then up to 3 more characters. Drive and path names may not be used here.

   In our example, you could type:

   SCHEDULE.AIR

   and press the RETURN key. SAM saves the schedule information in a file in your Work on Files diskette/directory and puts you back online with the service.

5. **Press the F2 key (Hang Up). Press the RETURN key.** SAM disconnects you from your service and displays your Service form.
6. **Press the F2 key (Return Form to Basket).** SAM displays the SERVICES basket.

10.4 Waiting For A Call

If your modem can automatically answer incoming calls, you can use SAM to receive a phone call from another computer. Before you begin, be sure that your modem is turned on, and correctly connected to your computer and the phone line. Then, look at the Desktop display, and press the F10 key (Wait for Terminal Call).

SAM will wait until a call is received, answer the call, and connect you to the calling computer.

10.5 Special Attributes

These are a series of communications options that allow you to customize SAM's exchanges with other systems.

10.5.1 Parity

Computer systems use parity to watch for errors in data transmissions. Different systems use different parity conventions; even parity and no parity are most common. As with the other special attributes, leave this set to the default value unless the service you will communicate with demands otherwise. No parity is always used to transfer files with the XMODEM protocol.

10.5.2 Data Bits

Different services dictate different numbers of data bits for each character sent or received. Leave this at the default setting unless your service demands otherwise. If you select NO PARITY, use 8 data bits.

10.5.3 XON/XOFF

Sometimes communications between different computers or systems need to be synchronized. By sending special control characters back and forth to indicate when to stop sending data and when to start again, the two systems can coordinate data transmission. The XON/XOFF protocol handles this task. You can also select hardware protocols, such as the RTS/CS.

The two parts of this section deal with the direction of data flow:

1. **Halt Incoming Data** Characters compatible with the remote service are set to start or stop incoming data.
2. **Halt Outgoing Data** The service you are communicating with may need to stop or start the data you are sending.

If data transmission is faster than 1200bps, then both incoming and outgoing data should probably use XON/XOFF to avoid data loss.

10.5.4 Incoming Data - Line Feeds

If the information SAM receives doesn't have line feed commands after carriage returns, it could be unreadable; each line would overprint the line before it. Though not normally necessary, you can set SAM to add those line feeds.

10.5.5 Incoming Data - Filtering Control Characters

Control characters (special codes that direct formatting of data) could make the data hard to read. The default setting instructs SAM to filter out control characters. When filtering is turned on the "high-order" bit of each incoming character is set to 0 and all incoming data is ignored except printable characters, carriage return, linefeed, formfeed, backspace, and bell. To capture all data "as-is", with control characters intact, choose the NO setting.

10.5.6 Outgoing Data - Pacing

Some computers cannot receive information as quickly as SAM can send it. This question asks if you want SAM to pause between each character sent. The default setting for pacing is .05 seconds; if messages you send appear to be missing characters, increase this number. (See Section 12.2 for more information on adjusting pacing.)

10.5.7 Outgoing Data - End of Line

This attribute controls what SAM does at the end of each line when sending a file. Most systems expect you to send a carriage return after each line. In this space you can type any other (macro) commands you want to execute at the end of each line. An example of a common choice is to add a delay at the end of each line:

```
send CR
wait 2
```

In this example, SAM will send a carriage return, wait 2 seconds, then send the next line of the file.

Or, if you need to wait for a prompt from the other system after each line, you might use something like:

```
send CR
wait 10 for LF "?"
```

In this example, SAM will send a carriage return, then wait for a linefeed, then a "?" from the other system before sending the next line in the file.

Notice the use of LF; this is because many systems echo back a carriage return and linefeed for the data you send to them. If your command just read "wait 10 for '?'", and your data contained a "?", the echo of that character could cause the next line to be sent too soon.

You can even tell SAM to let you manually send each line by using:

```
send CR
prompt "Press ENTER to send next line"
```

Here SAM sends a carriage return, then prompts you to press <Return> before sending the next line.

NOTE: These options do not affect file transfers using XMODEM.

10.5.8 Outgoing Data - Return (Enter)

The final question offers a similar choice; you can redefine what the Return (or Enter) key will do. The default, send a CR, works in almost all cases. However, if you are connecting to another terminal rather than a dial-up service, you may need to use "send CR LF". Again, you should leave this setting at the default unless it doesn't work with your system.

10.6 VT-100 Terminal Emulation

Many large computers use a certain type of terminal called the VT-100[10]. When you communicate with one of these computers, it may expect your computer to behave like a VT-100. If this is the case, SAM is prepared to help you.

Each Service form has an item entitled "Terminal Emulation". The item has two choices:

- **VT-100** lets your PC act like a VT-100 terminal when communicating. Most of the VT-100's special functions are available.
- **None** lets your PC act like a "dumb terminal" when communicating. It responds only to the most common terminal commands, and does not perform any special functions.

If a service requires a VT-100 terminal:

1. **Open the service's Service form.**
2. **Change your answer to the Terminal Emulation item from "None" to "VT-100" if necessary.**

When your PC is emulating a VT-100, most of your PC's keys act the same as they do normally, including the Function Keys (F1 - F10). However, since your PC does not have all of the keys that the VT-100 does, some VT-100 keys are imitated by using special combinations of the PC's keys, as shown in this table:

[10]VT-100 is a trademark of Digital Equipment Corp.

To get the VT-100 key:	**Press these keys on your PC:**
Del	Shift-Del (or Ctrl-Backspace)
Up Arrow	Shift-Up Arrow (or Alt-7)
Down Arrow	Shift-Down Arrow (or Alt-8)
Left Arrow	Shift-Left Arrow (or Alt-9)
Right Arrow	Shift-Right Arrow (or Alt-0)
PF1	Alt-1
PF2	Alt-2
PF3	Alt-3
PF4	Alt-4
Keypad Enter	Keypad +
Keypad ,	Alt-6

You can get all other VT-100 keys, including keypad keys, by pressing the corresponding key on your PC.

SECTION 11. WORKING ON FILES

The Work on Files function provides access to two general functions:

- Managing files and file directories
- Editing text files

Each function is associated with a particular SAM display. You manage files and file directories while looking at the File Directory display. You edit a file in the Revise File display.

To reach the Work on Files function, begin at the Desktop. Press the F9 key (Work on Files). SAM displays the first Work on Files display--the File Directory display.

11.1 File Directory Display

The File Directory display shows the files and sub-directories on your Work on Files diskette or directory. It is similar to the DOS command DIR. See Figure 11-1.

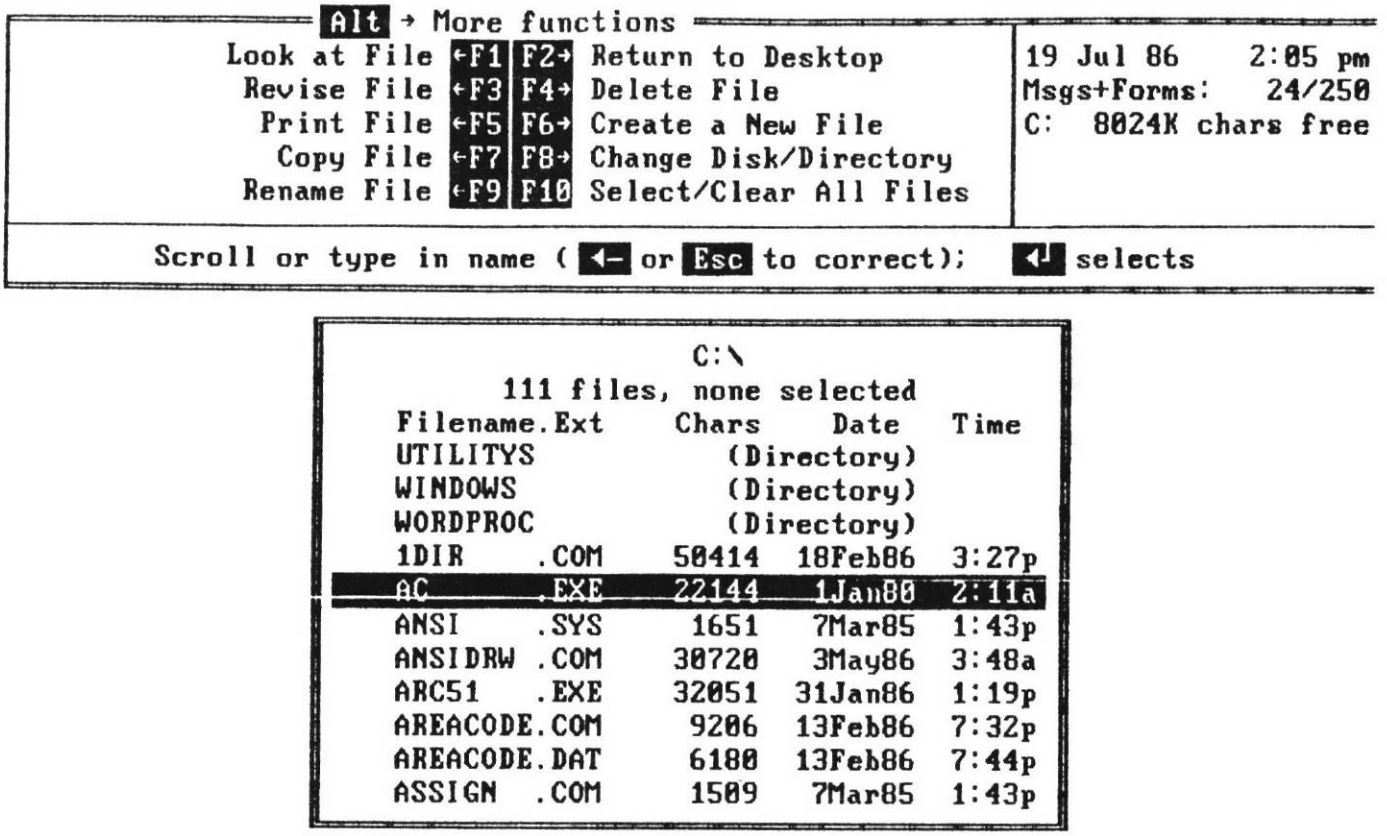


Figure 11-1: File Directory Display

The File Directory display gives you access to commands available in DOS without leaving SAM You can:

- Change disk drives or directories

- Look at directories or files
- Create directories or files
- Remove directories or files
- Revise files
- Print files
- Copy files from one drive and directory to another
- Select several or all files for multiplecopying, printing, or deleting

You can also perform these and other DOS functions by using the Execute DOS command. To execute a DOS Command, press Alt-F10 (both keys simultaneously). (Your PC must have at least 320k memory to use this function.)

The File Directory display shows each file's name, extension, number of characters, and date and time of its last update. If there are sub-directories on the disk, the directory name will be followed by:

(Directory)

Use the keys on the numeric keypad to locate the files listed in the File Directory display. You can also jump directly to a file by typing its name.

Though the File Directory display always starts by displaying your Work on Files drive/directory, you can switch to any other drive or directory by using the Change Disk/Directory function. Hold down the Alt key and press the F1 key. This does not change the default Work on Files drive/directory.

Important: When you communicate online, SAM will save all files retrieved from the other system in the Work on Files drive/directory. The Work on Files drive/directory is specified on the PC Configuration form in the YOUR PC basket.

Note: SAM will not let you operate on any file, such as TX0.FRM, that is part of the SAM program. This protects SAM from accidental damage.

While using the File Directory display, the Function Keys have the following definitions:

F1 - Look at File To perform this function, scroll to the file you wish to view. Then press the F1 key (Look at File). The file now appears in the lower part of your screen. Use the scrolling keys to move up and down in the file. When you're finished, press the F2 key (Return to Directory). While looking at a file, you can also choose to revise or print the file.

F1 - Look at Sub-directory
This is the same as Look at File above, except it appears when the highlighted bar is at the name of a sub-directory. Instead of looking at an individual file, you look at the sub-directory. For example, if your drive B directory contains a sub-directory called SOURCE, this function displays the files held in the SOURCE sub-directory.

F2 - Return to Desktop
Pressing this key takes you out of the File Directory display and back to the Desktop display.

F3 - Revise File
This function lets you revise any file. If the file is not a text file you will be warned in the Help window. For more information, see the following section.

F4 - Delete File

To perform this function, select the file you wish to delete. Then press the F4 key. The help line asks you to confirm the deletion. Type "Y" and press the RETURN key to confirm or press the ESC key to cancel. This function is the same as the DOS command delete.

Warning: When you delete a file using this function, it cannot be recovered.

F4 - Remove Directory

This appears only when your cursor highlights a directory. If you try to remove a directory that holds files, the help line will display an error message and prevent the removal. Before you remove an entire directory, use the F1 key (Look at Sub-directory), the F10 key (Select/Clear All Files) and the F4 key (Delete => Files).

F5 - Print File To perform this function, select the file(s) you wish to print, press the F5 key, then choose to print with headings (H) or without headings (RETURN). Print File prints the entire contents of the selected file(s). Make sure the file(s) you have selected are text files-- otherwise you may get unreadable printouts. SAM prints the file "as is". Print File With Headings prints the file name, date created, and page number at the top of each page. SAM starts a new page every 58 lines.

F6 - Create a New File

This function causes SAM to display an empty file screen. Use SAM's editor to type in text. Then save your new file. For information about file editing, see the following section.

F7 - Copy File Similar to the DOS COPY command, this function copies one or more files. Move the cursor to the file you wish to copy and press the F7 key.

The help line asks for the name of the destination file or directory. To copy the file to the same directory, type a name for the new file. To copy to a different drive or directory, type:

x:\directory

where "x" is the destination drive letter, and "directory" is the name of the destination directory. Then press the RETURN key.

To copy more than one file, move the highlighted bar to each file name and select it by pressing the RETURN key, so that there is an arrow pointing at it. When you've selected the files, press the F7 key. Type in the drive letter and directory name to which you want to copy the files. Then press the RETURN key. When you copy multiple files, you can't change their file names--you must copy them to another drive and/or directory.

F8 - Change Disk/Directory

This function switches disk drives and directories within drives. To perform it, hold down the Alt key and press the F1 key. The help line asks you for the new "Drive/pathname". Type the drive letter, colon (:), and if you want a specific sub-directory, a backslash (\) and the directory name. For example, to switch to drive B, type:

b:

To switch to a sub-directory called SOURCE in drive B, type:

b:\SOURCE

SAM displays the File Directory display, showing sub-directory SOURCE.

F9 - Rename File To perform this function, select the file you wish to rename and press the F9 key. The help line asks you to type in the new file name. Type in the new name and press the RETURN key.

Alt-F2 - Parent Directory

This function appears only when you are looking at a sub-directory. If you look at a sub-directory called SOURCE in your drive B directory, this function would take you back to the drive B directory.

Alt-F6 - Make Directory

This function creates a new sub- directory. Hold down the Alt key and press the F6 key. The help line asks you to type in the new sub-directory's name. Type the name and press the RETURN key.

Functions that accept directory names follow the DOS rules for path names. For example:

\directory	Refers to a directory that is a subdirectory of the "root" directory on the current drive. If C:\ACCESSII is the current directory, "\SOURCE" would refer to the directory C:\SOURCE.
directory	Refers to a directory that is a subdirectory of the current directory. If C:\ACCESSII is the current directory, "SOURCE" would refer to the directory C:\ACCESSII\SOURCE.

Alt-F10 - Execute DOS Command

This function lets you execute any DOS command without leaving SAM. To perform it, hold down the Alt key and press the F10 key. The help line asks you to enter a DOS command. Type the command you wish to use, such as DIR or COPY, and press the RETURN key.

You may also press the RETURN key without typing a command. This places you in DOS so that you can execute a sequence of DOS commands. To return to SAM, type EXIT.

The Execute DOS Command function is also available from the Desktop screen.

NOTE: You must have 320k memory to use this function. Insufficient memory may cause your PC to "lock up" and require turning it off to re-boot. It is best to try out this feature before doing critical work that might require it.

11.2 Revise File Function

Use SAM's Revise File function to revise files you receive from a service, prepare files to be sent out, or revise other text files.

To begin editing, select the File Directory display that shows the file you want. Move the cursor to the file name and press the F3 key (Revise File). SAM displays the file in revise mode.

The file editor works like a word processor. You can add, erase, and edit as many lines as you like, up to 64,000 characters, or about 32 full 8-1/2 x 11 inch pages, in any format you like.

When you Revise File, SAM shows you only printable (ASCII) characters. If the file also contains control characters (binary data), SAM strips these away before you begin editing. For example, SAM would strip the formatting characters (^B, ^S, etc.) from a WordStar document and leave only the text. The help line warns you when this happens.

The cursor is a small blinking underline or square. When you strike a key, you place a character at the position of the cursor. When you reach the end of a line, SAM will start a new line automatically.

For a complete explanation of all the editing keys and functions, see Section 4.3.

11.2.1 Insert and Overstrike Modes

When you begin to Revise Message, you are in Insert mode. In Insert Mode, the cursor is a blinking square, and as you type, your new characters will push the existing characters to the right. Press the Ins key (the white key on the lower right of your keyboard) to switch to Overstrike mode. In Overstrike mode the cursor is a blinking underline, and as you type, your new characters will replace existing ones. To return to Insert mode, press the Ins key again.

Generally, Overstrike mode is more useful when editing tabular information, while Insert mode works better when editing text in paragraph form.

In the following example shows how you might edit the file retrieved in the example in Section 10. SAM provides the same editing functions you used to write a message.

Before you begin, you should be looking at the Desktop display. Then, follow these instructions:

1. **Press the F9 key (Work on files).** The File Directory display appears.
2. **Move the cursor to the file name you just created.** (Example: SCHEDULE.AIR)
3. **Press the F3 key (Revise File).** The screen displays the file you retrieved from your service.
4. **Experiment with the editing keys to familiarize yourself with their functions.** Rearrange lines. Erase irrelevant words. Try pressing the F7 key (Delete Word Left). Press the F9 key (Delete Line), and then press the F10 key (Restore Line). Press the Ins key to turn Insert Mode on and off. You will notice the cursor change from an underline (_) to a box and back to indicate overstrike and insert modes. See Section 4.3 for more information on editing.
5. **When you're finished, press the F2 key (Save File).** Press the RETURN key to save the file under the same name.
6. **Hold down the Alt key and press the F2 key (Exit to Desktop).** The Desktop display appears.

SECTION 12. FILE TRANSFERS

Sometimes you will have a file that you want to upload to your mail host or transfer to another PC. This section will discuss the various ways to accomplish such file transfers.

Note that if you are connecting to your host through a DDN TAC, there are some special considerations (discussed in Section 12.4 below).

12.1 File Types

There are two types of files you may want to transfer from one computer to another, text files (those containing only printable characters) and binary files (those containing 8 bit characters, such as programs, spreadsheets, etc.). There are also two ways of transferring files, with verification (error checking) and without.

Text files can be transferred either with or without verification.

Binary files can only be transferred with verification.

To send files to another computer you should use the Interactive Terminal Service. This service has function keys setup to handle both types of file transfers.

12.2 File Transfer Without Verification

This method of transfer can only be used with text files. When selected, it sends the file as a stream of characters to the other computer, just as if you were typing them on the keyboard. To send a file using this method:

1. **Connect to the other computer using the Interactive Terminal Service.**
2. **Start the program on the other computer that will receive the file.** This may be a text editor, a mail program, or just a command to put all text received into a file.
3. **Press F10.** (Send File - no verify). This key tells SAM that you want to send a file without verification. The "Work on Files" window will appear. If necessary you can change directories to the one where the file resides. Select the file by moving the cursor-bar down to it (use the arrows, or type the first few letters of the file's name) and marking the file with the Return key.
4. **Press F2** (Done Selecting). SAM will close the Work On Files window and immediately begin sending the file.

If your host system has trouble receiving the file as fast as SAM sends it, increase the Pacing variable in your Interactive Terminal Service form (.05 is a good delay to start with). You can do this while online by pressing Control-F3, which will let you edit the service currently in use. Changes you make will take effect as soon as you resume online. Note that the pacing variable effects all information SAM sends, including keystrokes, so you may want to set the delay back to zero when you have finished.

To receive a text file without verification:

1. **Type the host command to type the file to the screen, but don't press Return yet.**
2. **Press F6** (Clear Buffer).
3. **Press F4** (Capture Buffer On/Off). This will save everything that appears on the screen in your PC's memory. NOTE: this assumes that Capture is already turned off. If it is on, you may omit this step.
4. **Press Return** to start the host typing the file to the screen.
5. **When the file ends, press F4** (Capture Buffer On/Off). This ends the saving of screen information in memory.

6. **Press F8** (Save Buffer). SAM will ask for the file name you want to use. Type in a file name, and press Return. Your file has now been saved onto your PC's disk.

12.3 File Transfers with Verification

To transfer a file with verification, both computers involved must be running a program to accomplish the required error-checking. To make the transfer, the program is started on both machines. The sending machine then breaks the file into small chunks (called "packets"), each of which contains information to verify that the packet was undamaged. The receiving machine gets the packets, checks that they were received without error, and reassembles them into the file.

Two commonly used protocols are the XMODEM (or MODEM) protocol, and the KERMIT protocol. SAM has XMODEM built-in, and lets the user run KERMIT by pushing to DOS using a function key.

12.3.1 XMODEM File Transfers

XMODEM was originally developed to send files between PCs. Some host computers now have a version available, which lets them exchange files with PCs. If you are connecting to your host through a TAC, be sure to read Section 12.4 below. To send a file using XMODEM:

1. **Start the XMODEM program on the other machine and tell it to receive a file.**
2. **Press F9** (Send File with XMODEM). The "Work on Files" window will appear. If necessary you can change directories to the one where the file resides. Select the file by moving the cursor-bar down to it (use the arrows, or type the first few letters of the file's name) and mark the file with the Return key.
3. **Press F2** (Done Selecting). SAM will close the Work On Files window and immediately begin sending the file. SAM will return to terminal mode when finished sending the file.

To receive a file using the XMODEM protocol:

1. **Start the XMODEM program on the other machine and tell it to send the file.**
2. **Press F7** (Receive File w/ XMODEM). The top line of the screen will display the message "Receive File with XMODEM; file name:".
3. **Type in the name you want the file to have on your PC. Press Return.** SAM will receive the file and return to terminal mode when finished.

12.3.2 KERMIT File Transfers

The KERMIT file tranfer protocol was developed at Columbia University. It is designed specifically for transfers between large computers and PCs. For this reason it is available on most host computers on the DDN.

At this time, SAM requires you to have a copy of the PC version of KERMIT to do KERMIT file transfers. MS-DOS KERMIT version 2.29 is included with SAM. The latest versions of KERMIT for all computers (both PC's and hosts) are public domain and are available at little or no charge from several sources. See Appendix C for more information.)

This section will not be a KERMIT tutorial. Before running KERMIT from within SAM, you should obtain KERMIT manuals and learn how to do KERMIT file transfers using just KERMIT.

To run KERMIT with SAM you will need to have:

- KERMIT software running on your host (check with your Host Technical Liaison).
- Information on how to use your host KERMIT.
- KERMIT software to run on your PC (provided with SAM).

- Information on how to use your PC KERMIT (provided with SAM[11]).

SAM makes it easy to use KERMIT by providing a several function keys to start KERMIT on your host, then suspend SAM, transfer to DOS, and run KERMIT.

Three keys in the Interactive Terminal service have been set up to run KERMIT.

- **F9 - Receive File w/ KERMIT** is used to download files from your host to your PC.
- **F7 - Send File with KERMIT** is used to upload text files from your PC to your host.
- **F5 - KERMIT Send 8-bit File** is used to upload binary files from your PC to your host.

These macros were set up to work with the TOPS-20 version of KERMIT. Since your host computer's KERMIT interface may differ from the TOPS-20 KERMIT, you may have to modify the macro commands. If you are running SAM from floppy diskettes, you will have to modify the macros to take this into account. For this reason, the "F5" macro is listed here and annotated line by line. (The text of the macro is underlined, with comments below each line.)

send 'kermit' CR **! start host KERMIT**

wait for 'Kermit' **! use your own host KERMIT's prompt**

send 'set receive packet-length 60' CR
This adjusts the packet length down to a size that the TACS can handle.

wait for 'Kermit'

send 'set file bytesize 8" CR **! put KERMIT in binary mode**

wait for 'Kermit'

send '~B O S' CR **! put TAC in binary mode**
send '~B I S' CR
These commands put the TAC in binary mode. The TAC must be in binary mode to use KERMIT, even if you are transferring text files. The DDN Network form changes the TAC intercept character to the tilde (~). If you are not connecting through a TAC you can put an ! in front of these lines to make SAM ignore them.

send 'receive' CR **! put KERMIT in receive mode**
This readies the host KERMIT to receive a file.
wait 1
The host KERMIT is now ready to receive the file.

!prompt 'Put KERMIT disk in drive A:, then press RETURN' length 0
If you are running SAM from floppy diskettes, delete the "!" in front of this command to activate it. It will cause SAM to pause so that you can replace the SAM PROGRAM disk with the SAM KERMIT disk.

---

[11]The manual for MS-DOS KERMIT is included as a text file, KERMIT.DOC, on your SAM SERVICES disk.

prompt 'Name of binary file to send to host: '
This line makes SAM ask the name of the file you want to send. Your answer is stored as a variable called "response", which is passed to KERMIT below.

show CR LF 'Starting MS KERMIT now...' CR LF

dos 'KERMIT send ' response ! push to DOS and start MS KERMIT
This command tells SAM to start DOS, then to run KERMIT and send the file name stored in the variable "response".

send 'quit' CR
Shut down the host KERMIT.

send CR
Get the host prompt back.

Figure 12-1: Sample KERMIT Macro

As you can see, when SAM does a KERMIT file transfer it is actually exiting to DOS and running a separate KERMIT program. Running KERMIT actually requires three programs:

1. KERMIT.BAT, the batch file that starts KERMIT
2. MSVIBM.EXE, the KERMIT program
3. KERMIT.INI, a configuration program that initializes KERMIT.

Depending upon your equipment, you may need to modify the KERMIT.INI file. It is a simple text file that contains commands which set up KERMIT's initial parameters. The file that comes with SAM looks like this:

```
set send packet-length 60
set receive packet-length 60
set baud 1200
set port 1
```

If your PC uses different settings (for example, uses the COM2 port rather than COM1), you will need to edit this file. It can be edited using SAM's built-in text editor. Just press F6 ("Work on Files") at the desktop. (If you are running SAM from floppy diskettes, MSKERMIT.INI will be located on the SAM KERMIT diskette. You will need to put this diskette in drive A before pressing F6.) See the KERMIT documentation (the file KERMIT.DOC on your SAM SERVICES diskette) for more information on customizing this file.

12.4 Binary File Transfer Through a TAC

TACs (and the DDN) were designed long before the current popularity of PCs. For this reason, verified file transfers through TACs are at present possible, but difficult. XMODEM works very poorly or not at all when going through a TAC. KERMIT will work in many cases if some TAC settings and KERMIT settings are adjusted. Some considerations are:

1. TACs normally use the "@" to precede a command (this is called the "intercept" character). SAM changes this to the tilde (~), so that addresses containing an "@" can be sent without problems.

2. TACs can only buffer 64 characters at a time. When using KERMIT you must reduce the send and receive packet sizes on both the PC KERMIT and the host KERMIT. Check your KERMIT documentation for instructions on how to do this.
3. TACs normally transmit only text (7 bit) characters. KERMIT requires the transmission of binary (8 bit) characters. The TAC must be put into the binary mode before using KERMIT. This is done by sending the TAC the commands "B I S" and "B O S", preceded by the TAC intercept character ("~" if you are using SAM; see example above) and followed by a Carriage Return. Note that not all hosts will accept binary mode.

If you adjust all these settings, and still cannot make successful file transfers through the TAC, contact your Host Technical Liaison. (Call the NIC at 800 235-3155 if you are unsure of who your Technical Liaison is. The NIC staff may also be able to help with KERMIT problems.)

SECTION 13. AUTO LOGINS AND KEYBOARD MACROS

When you communicate with information services and other systems, you often perform a few important tasks over and over. SAM provides automated functions to accomplish some of these tasks. For example, the F6 key sends and receives your mail. But other tasks may be unique to your needs.

You can create Auto Logins and Keyboard Macros to automate many unique tasks. Instead of typing all the commands required by a service to complete a given task, you press a single function key and SAM enters the commands for you.

To connect to a service, you must type a particular sequence of commands required by that service. SAM can record those login commands. As you type, SAM places the commands in the Auto Login portion of the Services form. The next time you connect to that service, SAM performs the login automatically.

An Auto Login is a particular kind of Keyboard Macro--one that logs you in to a service. It differs from other Keyboard Macros because you don't assign a particular function key to it. You simply call the service by pressing Dial a Service.

To learn how to record an Auto Login, go to the Automatic Learn Mode section, Section 13.3.

13.1 What is a Keyboard Macro?

A Keyboard Macro is a sequence of commands you perform by pressing a single function key. You use it while you communicate with an information service or another computer system.

Whenever you add a new service, SAM automatically assigns standard functions (or macros) to the function keys.

You can erase all predefined functions except F2 (Hang Up), and create your own Keyboard Macros or--as we recommend--leave at least some standard functions, such as F8 (Save Buffer), F6 (Clear Buffer), and F5 (Print On/Off).

For each service you communicate with, you can build a different set of macros. Use SAM's Automatic Learn Mode to record the commands that make up one macro. SAM records your commands in the Keyboard Macro portion of the Service form. Each macro can contain as many commands as you want--as long as the whole Service form doesn't exceed 20,000 characters.

13.2 Advantages of Auto Logins and Keyboard Macros.

When you use a service, you have to type commands to:

- Log in
- Find a certain area of information
- Transfer the information from the service to your computer
- Perform various other operations
- Log off

You pay the service and communication charges for the time it takes to enter each command manually. Your online session is costly, tedious, and error- prone.

Auto Logins and Keyboard Macros can accomplish the same tasks automatically. You spend less time communicating online and you get the same data.

Also, an experienced user or system administrator can prepare Auto Logins and Keyboard Macros for less sophisticated users. Using Auto Logins and macros, you'll make fewer mistakes and require less training time.

The sample comparison on the following pages shows what a Keyboard Macro does. This example assumes you've already recorded the interaction shown in the left-hand column and saved it as an Auto Login and Keyboard Macro.

Interaction Using Auto Login and Keyboard Macro:	Same Interaction Performed Manually:
With Auto Login:	Without Auto Login:
Press the F8 key (Dial a Service	Press the F8 key (Dial a Service). The screen prompts: Connected to THE SOURCE You type: "ID", your user ID, and password. Then press RETURN.
THE SOURCE displays its main menu.	THE SOURCE displays its main menu.
With Keyboard Macro:	Without Keyboard Macro:
Press the F4 key (SF-NY FLIGHTS).	The screen prompts: Enter item number You type: 6 and press RETURN The screen prompts: -> You type: AIRSCHED-D and press RETURN The screen prompts: Domestic Air Schedules You type: San Francisco, CA / New York, NY
The screen displays air schedules between SF and NY	The screen displays air schedules between SF and NY.

13.3 Automatic Learn Mode

Automatic Learn Mode allows you to record those portions of your online session you wish to save for re-use. SAM records:

- The commands you send to the remote system
- The questions, or prompts, the remote system sends to you

and places the recorded sequence into the Service form. SAM also remembers the predefined functions, such as Print On/Off or Save Buffer, which you used during your session.

There are two ways to use Automatic Learn Mode. First, you can set up to record an Auto Login as you connect to a service. Follow these steps:

1. **In the SERVICES basket, select a service and press the F3 (Revise Service) or F4 key (Add New Service).** SAM displays the service form. If you are adding a new service, type in the service's phone number before proceeding.
2. **Press the F1 key (Dial this Service).**
3. **Type "L" to turn on Automatic Learn Mode.**
4. **Log into the service.**
5. **When you've finished the sequence of commands, hold down the Ctrl key and press the End key.**

Second, you can use Automatic Learn Mode to record a Keyboard Macro once you're communicating online. Follow these steps:

1. **Begin an online session by pressing Dial Service (from the SERVICES basket or a Service form).**
2. **When you're online and ready to record, hold down the Ctrl key and press the F6 key.** SAM displays the function keys.
3. **To record a Keyboard Macro, press the function key you want to assign to that macro.** For example, press the F1 key, which is presently blank on most of the Service forms. To record on the second set of function keys, hold down the Alt key and press the function you want to assign to that macro. Do not record over the F2 key (Hang Up).

   The cursor is then positioned inside the label of the function key you wish to record. If that key already has a macro assigned to it, your new macro will replace the old one.
4. **Type in the new macro name and press the RETURN key to save the name and start recording.** To cancel, press the Esc key.

   SAM puts you back online (the function key panel disappears). The help line says that you're recording a macro. For example:

   RECORDING F4: Macro Name
   Ctrl-End: Stops Recording
5. **Type in the commands that perform the desired task.** SAM records the service's prompts and your typed commands in the Keyboard Macro portion of the Service form you are using. They appear in the form below the function key you selected.

   Where you type a command, SAM writes in your command in quotes:

   send "your command" CR

Where a system responds (usually following each SEND), SAM puts in a WAIT FOR command, with the system's response in quotes:

wait for "system response"

Press functions keys as needed. SAM writes these commands into your macro. You can even press a function key that invokes macros you have previously created. SAM copies these commands into the current macro.

6. **When you've completed the task, hold down the Ctrl key and press the End key to stop recording.** When you log off, the Service form reappears. You can edit the new Keyboard Macro or Auto Login as needed. Then press the F2 key (Return Form to Basket).

   The next time you use this Service form to log in, you have available the Auto Login or Keyboard Macro you recorded.

13.4 Testing Macros

After you record the Auto Login or Keyboard Macro, check that it does what you wanted it to do. Print out the Service form so you can see the macro commands. Then go back online. Next, follow these steps:

1. **Press the function key(s) that execute the macro.** Compare the action on the screen with your macro and note any unexpected behavior. The status line at the top of the screen shows which macro command is being performed.
2. **If the macro doesn't do what you wanted, hold down the Ctrl key and press the F3 key.** Do this after the macro has finished--or failed--but while you're still online. If the macro gets "stuck", press the Esc key to stop it.

   Edit the macro using the commands shown at the end of this section. When you are done, press the F2 key (Resume Online).
3. **Execute the macro again until you're satisfied.**

While editing a macro, you can press the F3 key (Macro Help). A small window appears on the right side of your screen. An additional function key appears to let you print the help information: F5 (Print Help Text). Macro Help gives you a list of macro commands, a glossary of terms, examples of macros, and other information you may find useful.

NOTE: Print out the Service forms already in your SERVICES basket to use as examples.

The easiest way to build your own macros is to use SAM's Automatic Learn Mode. You can also edit your own macro commands after you've created them, or type them in manually in the appropriate Service form.

You can edit while online, but you may want to wait until you're offline to do extensive editing, to minimize the time you are connected to the service.

To edit a macro while offline:

1. **Go to the SERVICES basket.**
2. **Move the cursor to the service for which you want to edit the macro.**
3. **Press the F3 key (Revise Service).**
4. **Select the Keyboard Macros option for the question:**

Do you want to setup or revise:
Auto Login
Keyboard Macros
Special Attributes
VT-100 Setup

5. **Move the cursor to the Keyboard Macro definition you want to change.**

While editing, look up the macro commands in the next section, or in the MACRO HELP window by pressing the F3 key. Type in the desired commands using the same editing keys you use when you revise a file.

When you save your Service form, SAM examines your macro with its built-in syntax checker. The syntax checker cannot tell if your macro will do what you want. It can check to see that your macro commands are properly spelled and include all the necessary information. If there is a syntax error, SAM puts the cursor at the location of the error and displays an error message in the help line.

Each macro command is on a separate line (paragraph). If you need to enter a single command that's longer than a full line, keep typing, and the cursor will wrap around to the left margin. This is still considered part of the same "line".

13.5 Macro Commands

The following list shows the special symbols and commands used in building and editing macros. You can also see this list by pressing the F3 key (Macro Help), while you are editing macros in a Service form. When you press Macro Help, a window opens on the right side of the display. The macro symbols and commands appear in that window. Use the cursor keys to scroll through the list.

Symbol	Definition
string	One or more of the following: • Text surrounded by single or double quotes, e.g. 'text text' or "text text" • Control code, e.g. ^Q, ^31 • Special symbol, e.g. CR, LF, or ESC • Variables (listed below) For example: ^X "id " response CR would be considered a string.
n	A number, e.g. 30 or 1.5
name	The name of a Network or Modem Driver form. Must be enclosed in quotes if the name contains spaces, e.g. 'my modem'.
[ ]	Indicates that the contents of the brackets are an optional item.
/	Separates choices.
on/off	Any one of the following: ON to turn a setting on OFF to turn it off TOGGLE to reverse a setting
variable	Any one of the following:

.sn, .f#, .n1, .ha, .id, etc.

Each of these variables is associated with a field in a Network or Service form. A complete list of variables appears in the "Macro Variable" column in each form. Each variable appears to the right of the field with which it is associated.

For example, .f# refers to the current phone number field, .ha refers to the current MM Host Address, and so on. You can use these variables in the access procedure portion of the current Network or Service form.

The variable names in your Service form will differ depending on whether you installed SAM Access with MM or Infomail.

In addition to the variables listed above, you can also use the following variables:

PHONE# Phone number from the current Service or Network form or ADDRESS BOOK entry

PULSETONE "P" or "T" as selected in the PC Configuration form

RESPONSE User input from the most recent PROMPT command.

In these command descriptions, UPPER CASE indicates command words that must be spelled exactly as shown. However, you may type any command in either upper or lower case.

13.5.1 Special Characters

You may need to enter special ASCII characters that are not printed on your keyboard or that already have an assigned meaning that SAM itself responds to. Characters not printed on your keyboard may include control characters (control Q, control S, control M, control J) used by the remote computer for special functions (XON, XOFF, carriage return, line feed).

When entering special characters on your keyboard that are also used by SAM itself (carriage return, escape), you need to make sure that SAM will not respond to them in the usual way. For example, if you press the RETURN key on your keyboard, the cursor goes to the left margin and down one line on your screen. If you need to send a carriage return to the remote computer, use the special name CR (or ^M). These special codes are always used outside quotes. Anything inside quotes is sent literally. For example:

send CR	sends a carriage return
send "CR"	sends the two characters C and R

There are as many as three ways to specify special characters:

Special Character	You Enter ASCII	You Enter Decimal	You Enter Command
Control A	^A	^1	-
Control B	^B	^2	-
.	.	.	-
.	.	.	-
.	.	.	-
Control Z	^Z	^26	-
Null	-	^0	-
ESC	-	^27	ESC
CR	^M	^13	CR
LF	^J	^10	LF
DEL	-	^127	-

13.5.2 The Commands

The commands listed below are the building blocks for keyboard macros. You can type commands in upper or lower case letters. Lower case is easier to read and type.

! ! may be used anywhere on a line, outside quotes, and the rest of the line will be ignored.

BAUDRATE [n] Sets specified baud rate. The allowed rates are: 110, 300, 1200, 2400, 4800, 9600 and 19200. If a baud rate is not specified, the rate in the current Service form is used.

BREAK [n] Sends a break code for n seconds to the remote computer (0.3 seconds if n omitted). BREAK is a special kind of interrupt signal used by many computers.

CAPTURE on/off Turns data capture on or off. If off, printing is also disabled. Capture saves incoming data from the remote computer in a disk file known as the "buffer".

CLEAR [BUFFER]
Discards current contents of capture buffer. As indicated, the word BUFFER is optional.

CONFIRM [string]
Displays optional string in help line, then asks you to press the RETURN key to continue the macro or the Esc key to cancel it. For example, the Hang Up key (F2) uses CONFIRM to let you change your mind if you hit the key by mistake.

DATABITS [n] Sets the number of bits in each data "word". The allowed values are: 5, 6, 7 and 8. If a value is not specified, the value in the current Service form is used.

DIAL [string] Dials the phone number specified in string. If string is omitted, dials the phone number found in the current Service or Network form or ADDRESS BOOK entry. Normally the first command in an Auto Login or Access Procedure.

DO string [parameters]
Performs the Access Procedure in the Network form named string. If string contains any blanks, the entire name must be put in quotes. Up to nine parameters may be specified. These parameters may be quoted strings or variable names such as PHONE#. Within the Network form, they are referred to as %1, %2, etc. through %9.

DOS [CLS] [commandstring] [PAUSE]
Executes a DOS command or sequence of DOS commands named commandstring. If you omit commandstring, DOS prompts you to enter commands until you type EXIT, which returns control to SAM. CLS clears the screen before running the DOS command. PAUSE causes the following message to be displayed when the DOS command is completed:

Press [Enter] to return to SAM

DTR on/off Controls "Data Terminal Ready" modem signal. Used within a Modem Driver.

END [SESSION] Ends the current online session, returning to the Desktop display. Does not hang up the phone. As indicated, the word SESSION is optional.

HANG [UP] Hangs up the phone. Does not end the online session. Use HANG UP followed by END SESSION to hang up and return to the Desktop display. As indicated, the word UP is optional.

[KEYBOARD] ECHO on/off
Controls local echo of keyboard input and "send file" data. Helpful if the remote computer is a half-duplex system, and therefore does not echo your characters. As indicated, the word KEYBOARD is optional.

MODEM string Following DIAL and WAIT FOR CALL commands will use the Modem Driver form named string. The string can be replaced with the RESPONSE command, allowing SAM to PROMPT you for the name of the Modem Driver form.

OUT1 on/off Controls "Out1" modem signal. Used within a Modem Driver to reset a Hayes internal modem.

PARITY [NONE/EVEN/ODD/ON/OFF]
Sets the specified parity. If none is specified, the parity in the current Service form is used.

PAUSE n Pauses n seconds. PAUSE waits only for the specified time, regardless of whether you are receiving data or not (See WAIT n).

PORT [COM1/COM2]
Selects serial communications port COM1 or COM2. If neither is specified, the port selected on the current Service form is used.

PRINT on/off Turns the printer on or off, to get a printo ut of what appears on the screen. CAPTURE OFF disables printing, even if PRINT ON is used. If you turn on CAPTURE and PRINT, printing will continue until you use PRINT OFF or end the communication session.

PRINT string Prints string.

PROMPT [HIDE] string [LENGTH n]
Displays string in help line, allows keyboard input of up to n characters, saves input as RESPONSE. If n omitted, assumes as large as possible. HIDE makes the keyboard input not display when typed in.

For example, you could use the command:

prompt hide "Password:" 8

SAM would make the string "Password:" appear in the help line, followed by brackets containing 8 spaces to be filled in. You would then type in up to 8 characters, which would display as "XXXXXXX" and press the RETURN key. The characters you entered would be saved under the name RESPONSE. The next line in your macro could be:

send response CR

This would send the characters you just entered to the remote computer.

RTS on/off Controls "Request To Send" modem signal. Used within a Modem Driver.

SAVE [BUFFER] [string]
Saves the contents of the capture buffer to a file named string. If string omitted, prompts for file name.

SEND string Sends string to the remote computer. The string may be a character, word, or phrase with or without special characters.

SEND FILE [string]
Sends file named string to the remote computer, obeying "end of line" commands. If string omitted, gives lookup window.

SHOW string Displays string on the screen and copies into capture buffer. Use this for reminder messages.

STOPBITS [n] Sets the specified number of stop bits. The allowed values are 1 and 2. If neither is specified, the value in the current Service form is used.

TRACE on/off Displays your macro commands in the help line of the online screen. TRACE OFF turns off the display. You might want to use TRACE OFF to hide your passwords in an Auto Login.

TRANSLATE variable string1 string2
Finds all occurrences of string1 in the variable, and changes each one to string2. The variable can be PHONE#, PULSETONE or RESPONSE. The strings can be of any length.

WAIT n Waits until n seconds go by with no data received from remote computer. Used when the remote computer does not provide a specific prompt for the next input.

WAIT [n] FOR string
Waits until either:

- n seconds go by with no data received from the remote computer (aborts macro). If n is omitted, 60 seconds is assumed.
- string is received from the remote computer (continues macro). SAM ignores case (upper-lower) but checks for space, CR and LF characters.

WAIT [n] FOR signal
Like WAIT FOR string, but waits for the specified modem signal:

CARRIER	Data Carrier Detect	
	DSR	Data Set Ready
	RING	Ring Indicator

Used within a Modem Driver.

WAIT [n] FOR CALL
Waits n seconds for an incoming call. Executes the "Set up for answering calls" Modem Command Macro in the Modem Driver form, and then executes the "Check for incoming call" Modem Command Macro. Waits for the latter macro's response.

Typically used in an Access Procedure right after a HANG UP, to handle a "call back" security system. For example,

wait for call

would wait 60 seconds (the default value for n) for an incoming call.

XRECEIVE [string]
Receives file named string with XMODEM protocol. If string omitted, prompts for file name. This option is useful only between PCs, or to a service that accepts XModem protocol.

XSEND [string] Sends file named string with XMODEM protocol. If string omitted, gives lookup window. This option is useful only between PCs, or to a service that accepts XModem protocol.

13.6 Errors In Macros

When you complete and save a keyboard macro, SAM checks it for syntax errors. If there is a syntax error (if a keyboard macro won't work), SAM puts the cursor at the location of the error and displays an error message in the help line. At the same time, you'll be prompted to press the Esc key.

Press Esc and correct the macro. Save the corrected version. SAM will check your command syntax again.

If SAM finds a syntax error while you're executing a keyboard macro, online interaction stops and you'll see an error message in the help line. Take the following steps:

1. **Press the Esc key.** SAM takes you off line, displays the Service form and positions the cursor at the location of the error.
2. **Correct the error by typing in the proper command syntax.**
3. **Press F2 (Resume Online). SAM takes you online again.**

These are the keyboard macro error messages:

Blank required before Quote mark
A quote mark was found right after a command (e.g. SEND"FOO"). There should be a space character separating commands.

Blank required after Quote mark
A blank is required after the closing quote of a quoted string. (Example: SEND "FOO"CR).

Command name required
Something other than a command name was found at the beginning of a line (e.g., a quoted string with no preceding command).

Invalid control character
A ^ control character was invalid (such as ^?).

xxx required here The keyword xxx is required here (e.g., KEYBOARD XYZ would say "ECHO required here").

Line should end here
Extra characters were found where the line should have ended. (Example: ECHO ON OFF should say only ECHO ON.)

Name or quoted string required.
The end of line was found instead of a required string. (Example: Send command with no string.)

ON, OFF, or TOGGLE required
End of line or another word was found where ON, OFF, or TOGGLE was required. (Example: PRINT XYZ.)

Quoted string not terminated
The ending quote (" or ') was not found before the end of a line.

Unknown command: xxx
The name xxx was found at the beginning of a line and is not a valid command.

Unknown name: xxx
The name xxx was found where a quoted string of CR, LF, ESC, or RESPONSE was expected and is not a valid command structure. (Example: Send "Hi" Send.)

13.7 Sample Auto Login And Keyboard Macro

The following examples show how an Auto Login and Keyboard macro were used to communicate with THE SOURCE. These examples perform the steps shown in the comparison of manual and automated interaction at the beginning of this section.

In the following Auto Login example, SAM dials a long-distance telephone company, connects to THE SOURCE, logs you in, and waits until THE SOURCE is ready to receive your commands. See Figure 13-1.

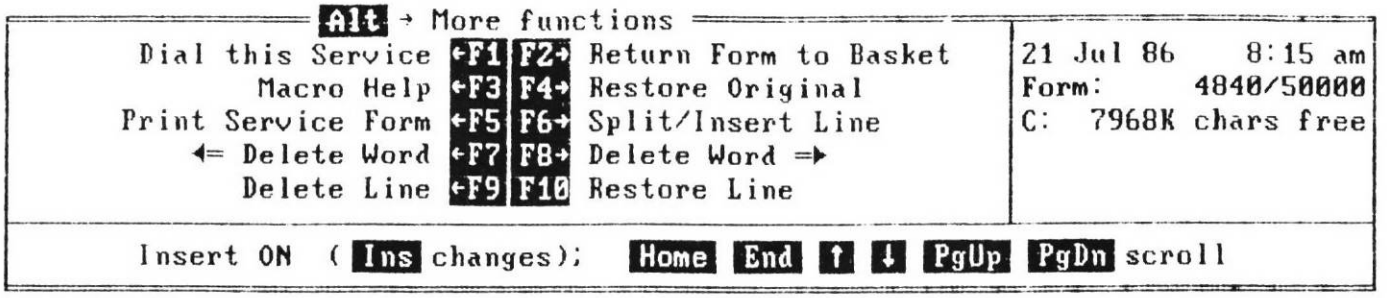


Figure 13-1: Sample Auto Login

The commands in the sample Auto Login do this:

dial Dials the phone.

wait for "Connected to THE SOURCE"
Waits 60 seconds for THE SOURCE's greeting, as specified.

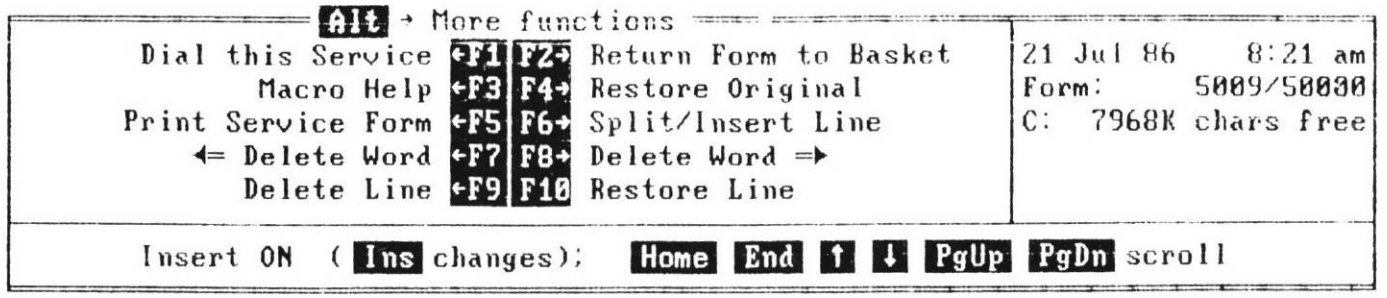
send "ID (your SOURCE user ID) (your SOURCE password)" CR
Sends the letters I and D, a space, your SOURCE user ID, another space, your SOURCE password, and a carriage return.

wait for "Enter item number"
Waits 60 seconds for the command prompt.}

After the Auto Login, THE SOURCE's main menu will appear, and you will be ready to enter your first command. In the following example, pressing the F4 key will show the schedule of airline flights between San Francisco and New York. See Figure 13-2.

The commands in the sample Keyboard Macro do this:

send "6" CR Sends the character 6, followed by a carriage return. Note that the character 6 is enclosed in quotes, since it is to be sent literally, while the special character CR is not enclosed in quotes.




```
K E Y B O A R D   M A C R O S

F1  Label:  SF-NY Flights
send "6" CR
wait for ^26
send "AIRSCHED-D" CR
wait for "Domestic Air Schedules"
capture on
clear buffer
send "San Francisco, CA / New York, NY" CR
wait 10
save buffer "SF-NY"
```

Figure 13-2: Sample Keyboard Macro

wait for ^26 Waits 60 seconds for the right-arrow prompt. On your PC, the right-arrow character is assigned the decimal value of 26. Therefore, ^26 signifies the right-arrow.

send "AIRSCHED-D" CR
Sends the word "AIRSCHED-D", followed by a carriage return.

wait for "Domestic Air Schedules"
Waits 60 seconds for this phrase.

capture on Turns on the capture feature. Everything you see on the screen will be put in the capture buffer, which is a temporary part of the computer's memory.

clear buffer Makes sure that the capture buffer is empty.

send "San Francisco, CA / New York, NY" CR
Sends this phrase, followed by a carriage return. After a few moments, the schedule of flights between San Francisco and New York will appear on your screen.

wait 10 Waits for 10 seconds to go by without receiving any data on the phone line. This indicates that the entire schedule has been received.

save buffer "SF-NY"
Saves the schedule in a disk file named "SF-NY".

You could do all of the preceding steps manually. But you would have to type each instruction and retype

it if you made a mistake. You would have to wait for the proper responses from the service and remember to save the information in a buffer and a disk file. By using an Auto Login and a single Keyboard Macro, you make the whole operation easy and error-proof.

SECTION 14. MODEM DRIVERS AND NETWORK ACCESS

This section shows how to customize SAM for use with various types of modems and special communications networks, including telephone systems, local area networks, special call-back security systems, and public data networks. Different modems and internal networks may have very different features and configurations. You can adapt SAM to nearly any situation by filling out the following forms:

- Modem Driver forms (in the YOUR PC basket)
- Network forms (in the SERVICES basket)
- Access Procedures (part of DDN Network forms).

14.1 Modem Drivers

Modem Driver forms tell SAM how to use your modem (see Figure 14-1). When you get SAM, the YOUR PC basket displays this list of Modem Driver forms:

Modem driver: Acoustic Coupler/Manual Dial modem
Modem driver: Direct Connection
Modem driver: Hayes Smartmodem 1200 (external)
Modem driver: Hayes Smartmodem 1200B (internal)

SAM is already set up to work with any of these modems. To use one of them, select its name from the list in the PC Configuration form, which is also in the YOUR PC basket. When you begin communication, SAM uses the Modem Driver form corresponding to the modem you selected. If your modem is not listed, try selecting the appropriate Hayes modem. If this does not work, try the other modem drivers. If you still have no success, you probably need to add a new Modem Driver form.

When you create a new Modem Driver form, SAM adds that modem to the list in your PC Configuration form. When you revise the PC Configuration form, you'll see it along with the other modems.

To add a new Modem Driver form, follow these steps:

1. **In the YOUR PC basket, press the F6 key (Add a Modem Driver).** SAM displays a sample Modem Driver form. See Figure 14-1. You can also add a new modem driver by using the F7 key (Copy form) to copy an existing modem driver that is similar to what you want.
2. **Fill in the form (described below).**
3. **Press the F2 key (Return Form to Basket).** SAM displays YOUR PC Basket with the new modem driver listed.
4. **Move the cursor to the PC Configuration form and press the F3 key (Revise Form).** SAM displays the PC Configuration form.
5. **Scroll to the question "What modem are you using?" and select the modem you just defined.**

Each form has the following sections:

Modem Name: The first line asks for "Name of Modem:". Enter a unique name that will identify this modem for you. (Presently limited to 16 characters.)

Comments: The second area, titled "Comments", lets you type in instructions or comments to which you can refer later on. SAM ignores these.

Maximum baud rate:

The third section asks you to select the maximum baud rate at which your modem will work. If you have a 300 baud modem, select 300; if you have a 1200 baud modem, select 1200, and so forth. If a certain service uses a lower baud rate than the maximum set here, select the lower rate on your Service form.

14.1.1 Modem Command Macros:

You need four modem command macros to control your modem:

- Dial
- Set up for answering calls
- Check for incoming calls
- Hang Up

The operation of the "Dial" and "Hang Up" macros are straightforward. They send the appropriate commands to the modem to dial and hang up the phone. The "Set up for answering calls" macro sets the modem in the proper state for answering the phone. Then SAM uses the "Check for incoming calls" macro repeatedly in a loop until a call is received.

In the Modem Driver form, enter the appropriate commands for your modem. To decide which commands to use, check your modem's manual, the following explanation of the driver shown in Figure 14-1, and the description of commands listed in Section 5. When you're done, press the F2 key (Return Form to Basket) to save the completed Modem Driver form.

The commands in the example shown in Figure 14-1 do this:

To Dial:

DTR on Sets the RS232 line DTR high so the modem knows the computer is ready to communicate.

send 'AT E1 Q0 V1 X1 SO=0' CR

Sends commands specific to the Hayes modem to set up Echo ON (E1), Quiet OFF (Q0), Verbose response (V1), Connect 1200 response ON (X1), and Do Not Answer The Phone (S0=0). Then send a carriage return.

wait .8 Waits for .8 seconds of quiet after the modem replies, "OK".

send 'AT D' pulse ' ' phone# CR

Sends the command to dial a phone number. 'AT D' specifies to DIAL. pulse provides the P or T to set up pulse or touchtone dialing. phone# is the phone number you are calling. CR sends a carriage return.

wait 30 for 'CONNECT'

SAM will now wait 30 seconds for the modem to respond with the word CONNECT meaning it is in data communication with a remote modem.

To set up for answering calls:

DTR on

send 'AT E1 Qo V1 X1 SO=1' CR

Does the same things as in "To Dial:" except S0=1 sets up the modem to answer the phone after 1 ring.

wait .8 Waits for .8 seconds of quiet after the modem replies, "OK".

To check for an incoming call:

Figure 14-1: Sample Modem Driver

```
                M O D E M    D R I V E R
------------------------------------------------------------------------
Name of modem:  Hayes Smartmodem 1200  (external)

Comments:

Smartmodem 1200 switch settings (or factory settings are OK):
    1, 6:   UP recommended (not required)
    7:      UP for standard phone line, DOWN for multi-line phone
    8:      DOWN
    2, 3, 4, 5:  n/a

------------------------------------------------------------------------
Maximum baud rate:
      300
 =>  1200
     2400
     4800
     9600
    19200
------------------------------------------------------------------------
Modem command macros:

To dial:
DTR on                                          !Set terminal ready
send 'AT E1 Q0 V1 X1 S6=6 S7=120' CR            !Initialize modem options
wait .8                                         !Wait for it to acknowledge
send 'AT D' pulse ' ' phone# CR                 !Dial the phone number
wait 120 for 'CONNECT'                          !Wait max. of 30 sec for connection


To set up for answering calls:
DTR on
send 'AT E1 Q0 V1 X1 S0=1' CR
wait .8


To check for an incoming call:
wait 1 for 'CONNECT'


To hang up:
DTR off                         !Set terminal ready false
pause .2                        !Give modem time to sense it
DTR on                          !In case DTR is defeated by switches, try cmd
pause 1
send '+++'                      !Put modem in command mode
pause 1
send 'AT Z' CR                  !Reset and hang up modem
wait .5
send 'AT S0=0' CR               !Disable auto answer
wait .8
DTR off
```

wait 1 for 'CONNECT'
Waits 1 second for the connection response signal from the modem.

To hang up:

DTR off Drops the RS232 signal DTR to hang up the modem.

pause .2 Pauses .2 seconds to give the modem time to sense the DTR signal.

DTR on Sets DTR high again in preparation for entering modem "Command" mode to reset the modem.

pause 1 Pauses 1 second to start the hangup command. This is called a guard band by Hayes.

send "+++" Send the standard escape characters which will be followed by another 1 second guard band.

pause 1 Pauses 1 second.

send 'AT Z' CR Sends the modem reset command to set all options back to their default states, then sends a carriage return.

wait .5 Waits .5 seconds for a response back from the modem indicating that it has received and executed the previous command.

send 'AT S0=0' CR
Instructs the modem not to answer the phone, then sends a carriage return. This is to prevent the modem from answering the phone when your computer is off.

wait .8 Waits .8 seconds for the modem's ACKNOWLEDGE.

DTR off Drops the RS232 DTR signal.

14.2 Network Forms

If you use a direct telephone line and standard access to services, you can ignore SAM's special Network forms and Access Procedure commands. In other words, if the connection between your PC and the service looks like this:

```
Your PC
   => Modem
      => Direct Telephone Line
         => Service Without Call-Back
               or Other Special Procedure
```

simply fill out your Service forms and ADDRESS BOOK entries and begin mailing or communicating online.

However, you may have a local area computer network (LAN) or a special telephone system such as a PBX. Also, some services have call-back procedures: you call them, they call you back, and then you begin the session. Further, in many European countries you must go through a Public Data Network before you can get access to a service. To handle these unique situations, you can use SAM's special macro commands to define your own Network forms and Access Procedures.

Network forms, held in the SERVICES basket, tell SAM how to handle special telephone and computer systems. Each form contains a sequence of macro commands that help SAM deal with a certain situation.

You should create a Network form if you use a:

- PBX
- Local Area Network

- Modem Pool
- Call-Back System with your Service
- Public Data Network

You may need several Network forms or only one, depending on your situation. For example, if you have a PBX system which requires different methods for calling within the building, outside the building, and long distance, you may need three Network forms, labeled:

Internal Call

External Call

Long Distance

On the other hand, you need only one Network form if you have a local area computer network and you always use the same procedure to get through the network to the modem.

Here's how Network forms work: First you fill out a Network form with the information and commands SAM needs for your situation. Then, at the appropriate point in the Access Procedure section of the DDN Automatic Mail form, or in the Auto Login section of any Service form, you type the word "DO" followed by the name of the Network form. When you dial the service or other PC, SAM reads the DO command, and executes the commands found in the Network form. Then, SAM continues whatever activity it was performing.

One Network form can even use another Network form. This way you can adapt SAM to a highly sophisticated communications network.

Why are Network forms necessary? Each Service form, including the DDN Automatic Mail form, accepts sequences of macro commands. Therefore, you could type the macro commands required by your situation into each one of your Service forms. But this would be terribly time-consuming. Instead, enter the required commands only once, in a Network form. Then, in the Access Procedure section of the DDN Automatic Mail form, and in the Auto Login section of each Service form, simply enter a DO command, which executes the commands in the Network form.

14.2.1 Adding a Network Form

To add a Network form to your SERVICES basket, follow these steps:

1. **In the SERVICES basket, press the F9 key (Add a Network).** SAM displays an empty Network form.
2. **Fill in the form (described below).**
3. **Press the F2 key (Return Form to Basket).** SAM displays the SERVICES basket.

After completing a Network form, you must enter the "DO network name" command in the Access Procedure section of either the DDN Automatic Mail form or another Network form, or in the Auto Login section of another Service form. A Network form is useless unless it is referred to directly by a Service form or indirectly through another Network form.

14.2.2 Filling In a Network Form

The Network form has only three areas to fill in:

Network name: Enter a unique name which identifies the Network form. For example, "Internal Call" describes an internal PBX system.

Phone Number: The phone number is optional. For example, if you use a modem pool that does the dialing for you (i.e. if your computer is not directly connected to a modem), do not include a phone number. The phone number can be up to 40 digits. Enter a number if it applies in your case.

Access Procedure: The last item asks you to enter the macro commands that perform the tasks required by your situation. The available commands are described in detail in Section 5. Press the F3 key (Macro Help) to look at a help window of macro commands. Also, refer to the following section.

14.3 Access Procedures

The DDN Automatic Mail form and each Network form has a section called Access Procedure. This section is designed to hold a sequence of commands, referred to as a "macro", which is executed before or during the standard login.

The Access Procedure can be as short as the single command DIAL. Or it can be more complex, containing many commands.

Any Access Procedure or Auto Login can use a Network form's Access Procedure by using the DO command. When SAM encounters a DO command, it executes the commands found in the Network form named by the DO command.

Your Access Procedure can begin before or after you dial, depending on the sequence you require. For example, if you go through a local area network to reach the modem, your Access Procedure or Auto Login might look like this:

do "lan"

dial

However, if you go through a public data network to reach a service (which must be done after dialing), your form might look like this:

dial

do "pdn"

14.3.1 Using Access Procedures

For each service affected by your special system requirements, you need to add one or more DO commands in the appropriate Service forms. Follow these steps:

1. **Be sure you have completed the Network form or forms you will using.**
2. **In the SERVICES basket, scroll to the desired Service form. Press the F3 key (Revise Form).** SAM displays the Service form.
3. **Move the highlighted bar to Access Procedure or Auto Login. Press the RETURN key.** The Access Procedure or Auto Login portion of the form opens up and the screen displays the current Access Procedure or Auto Login (often simply the command DIAL).
4. **If necessary, revise the current Access Procedure or Auto Login. Then move the cursor to where you want the DO command, and type:**

   do

   followed by the name of the desired Network form. You may add as many DO commands as you wish.

5. **Press the F2 key (Return Form to Basket).** SAM displays the SERVICES basket.

14.3.2 Access Procedure Command Sequences

The following example shows an Access Procedure for a system that employs a call-back scheme. SAM dials the service, logs in, hangs up, waits for the call back, and answers the phone:

```
dial
wait 20 for CR
echo on
wait 20 for "Hello!"
wait 1
send 'your id' CR
echo off
hang up
wait 300 for call
```

The commands in the example do this:

dial	Dials the phone using the modem selected in your PC Configuration form.
wait 20 for CR	Waits 20 seconds to receive a carriage return from the remote system, then proceeds to the next macro.
echo on	Turns keyboard echo on.
wait 20 for "Hello!"	Waits 20 seconds to receive "Hello!" from the remote service.
wait 1	Waits one second, then proceeds to the next macro.
send 'your id' CR	Sends your system ID, then sends a carriage return.
echo off	Turns keyboard echo off.
hang up	Hangs up the phone.
wait 300 for call	Waits 300 seconds for the call back from the remote system.

14.4 Macro Commands

Section 5 contains a complete description of all macro commands. You can also see the entire list of commands by pressing the F3 key (Macro Help) while revising most forms. When you press Macro Help, a window opens on the right side of the screen. The macro symbols and commands appear in that window. Use the cursor keys to scroll through the list.

Not all commands can be used in a Modem Driver or Access Procedure. In general, commands that refer to disk files or printing are not allowed. See the list on the following page.

Allowed	Not Allowed
! comment	CAPTURE on/off
BAUDRATE [n]	CLEAR [BUFFER]
BREAK [n]	DOWNLOAD [folder.msg]
CONFIRM [string]	END [SESSION]
DATABITS [n]	PRINT on/off
DIAL [string] *	SAVE [BUFFER] [string]
DO network [parameters]	SEND FILE [string]
DTR on/off	UPLOAD [folder]
HANG [UP]	XRECEIVE [string]
[KEYBOARD] ECHO on/off	XSEND [string]

MODEM [string] *
OUT1 on/off
PARITY [NONE/EVEN/ODD/ON/OFF]
PAUSE n
PORT [COM1/COM2] *
PRINT string
PROMPT string [LENGTH n]
RTS on/off
SEND string
SHOW string
STOPBITS [n]
TRANSLATE variable
string1 string2
WAIT n
WAIT [n] FOR string
WAIT [n] FOR signal
WAIT [n] FOR CALL

* DIAL, MODEM and PORT are allowed in an Access Procedure, but not in a Modem Driver.

SECTION 15. MERGEMSG (MERGE MESSAGE PROGRAM)

MERGEMSG (Merge Message) is a program that can combine the messages, Address BOOK entries and Service forms from two copies or versions of SAM into one. You will find MERGEMSG on the SERVICES diskette.

You can use MERGEMSG when you are switching to a new version of SAM. You can transfer your old messages, addresses and forms to the new version; you don't have to redo the forms or rewrite the messages or addresses.

You can also use MERGEMSG to share your messages, addresses or forms with another SAM user.

Finally, if your copy of SAM is damaged, you can use MERGEMSG to copy old messages, addresses, and forms into a new copy of SAM. (If you have problems doing this because the file TX.DIR is damaged, delete the TX.DIR from the old SAM.)

Although MERGEMSG shouldn't cause any loss of messages when used correctly, **be sure to make a copy of your message diskettes or SAM subdirectories running MERGEMSG.** That way, if you make a mistake while using MERGEMSG, you can start over.

MERGEMSG adds the contents of one Message diskette or directory (Source) to another (Destination). The Destination diskette or directory may be newly installed or may already have messages in it. The messages, addresses and forms will be copied from the Source to the Destination in the same basket position.

MERGEMSG can copy an item into a basket that already has another item with the same name. When this happens, the two items are always listed consecutively when you look at that basket. The lower item is the one that was originally in the basket.

Follow these instructions to use the MERGEMSG program:

1. Insert the SAM SERVICES diskette in drive A. After the A> prompt, type:

   mergemsg

   and press the Return key. Follow the instructions on the screen.
2. Remove the SAM SERVICES diskette from drive A.
3. If your PC does not have a hard disk, place the Source Messages diskette in drive A. Then, insert your Destination Message diskette in drive B.
4. If your PC has a hard disk:
   - If both the Source and Destination messages are on the hard disk, skip this step.
   - If either the Source or Destination messages are on a diskette, insert the diskette in drive A.
   - If both the Source and Destination messages are on diskettes, follow the instructions for a PC without hard disk.
5. After the prompt, "Enter the SOURCE message disk/directory":

   If your PC does not have a hard disk, type:

   a:

   and press the Return key.

   If your PC has a hard disk, type the name of the drive and the directory containing the Source messages, and press the Return key.

6. After the prompt, "Enter the DESTINATION message disk/directory":

   If your PC does not have a hard disk, type:

   b:

   and press the Return key.

   If your PC has a hard disk, type the name of the drive and directory that contains the Destination messages, and press the Return key.

7. After the prompt, "Merge All or Selected basket(s)":

   If you want to merge the contents of every Source and Destination basket, type:

   a

   MERGEMSG will now merge all of your baskets.

   If you want to merge only the contents of certain Source and Destination baskets, type:

   s

   MERGEMSG will name each basket, one at a time, and ask you whether you want to merge that Source and Destination basket.

8. When MERGEMSG is done, it will say:

   Messages and forms successfully merged

If your PC has a hard disk, you are now ready to run your new SAM. If your PC does not have a hard disk, remove the Source Message diskette from drive A and replace it with the SAM Program Copy diskette. Then you are ready to run SAM.

When you run SAM, you will first need to re-do the PC Configuration form, because its contents are not copied. Note that there will be two of each modem driver, both the originals and those added by MERGEMSG. Delete the duplicates.

The one form that won't be merged is the Automatic Mail form. You will need to manually copy any changes you have made in the login procedure or macros, and to fill in your account name, password, etc. The standard network services (WHOIS, TACNEWS, and BIBLIO)) will also be doubled. Delete one of each, either the old or the new.

SECTION 16. TROUBLESHOOTING

SAM may not always work the way you expect it to. When this happens, review the current display; it will usually help you see what is going on. If you are confused, or do not know what to do, check the following list of problems and solutions.

NOTE: Many SAM problems can arise because of insufficient memory or conflicts with memory-resident programs such as Sidekick. If you are running a memory-resident program and experience problems, try removing the program then running SAM again.

Problem **Solution**

Messages I send are missing characters.

Adjust the pacing, found in the Special Attributes section of your DDN Automatic Mail form. Try a starting value of .05, and increase or decrease according to your local conditions.

I can't send mail Check your modem:

- Is it plugged into the right port on your PC?
- Is it turned on?
- Are its switches set correctly?

I'm trying to connect to other services and can't

Check the Service form:

- Do you have the correct login information?
- Do you need to adjust the baud rate?

SAM didn't log on to the host successfully

Check your Automatic form:

- Is your password and ID correct?
- Is the telephone number correct?

Read the login message in the IN basket.

My PC-AT doesn't complete the login script for the predefined services.

If you are using a PC that runs faster than the standard IBM PC, you will probably need to increase the values used for "WAIT" statements in the automatic logins. This will compensate for your machine's speed.

When I press the keys on the numeric keypad, my PC beeps

Press the Num Lock key and try again.

SAM doesn't receive my mail

Press the F6 key (Send & Receive Mail) to check the Mail Delivery Options form. Under the second item, "Send mail, receive mail, or both?" make sure that you have chosen either "Both send and receive mail" or "Receive mail ONLY".

I can't make the printer work

Look at the PC Configuration form in YOUR PC basket. Check your answers to the questions about printers.

Make sure that your printer is turned on, connected to the PC, and ready to print.

The Help window says: User number rejected

Look at the DDN Automatic Mail form in the SERVICES basket. Make sure that your User ID and password(s) are correct.

The Help window says: Sorry. Call did not go through

Is your modem turned on and connected to both the PC and the phone line?

The Help window says: Failed to reach Network

All of the DDN phone lines are busy. Put more than one phone number in your DDN Automatic Mail form. SAM will try each, and if it is busy, go on to the next.

The Help window says: Directory Full or Disk Full

You hard disk or floppy diskette is full of information. Shred a few items in the WASTE basket.

The Help window says: Drive--not ready. Correct and press the RETURN key.

Look at the PC Configuration form in the YOUR PC basket. Go to the bottom of the form. Check the answer for the last item, "Disk/directory to be used for Attachments and Terminal files:"

If your PC has a hard disk, the answer should be:

c:\directory

In place of "directory", you can put the name of any directory on your hard disk, or none at all.

If your PC does not have a hard disk, the answer should be:

a:\

The phrase "# of chars. free" is highlighted or flashing in the Status Display window.

There is 20K or less space on the diskette or hard disk. Shred a few items in the WASTE basket.

I am communicating interactively with my host, but I can't see what I am typing.

Press the F3 key (Echo On/Off).

SAM sent a message via MILNET/MAIL, but I found this message in the OUT basket: Rejected by MILNET/MAIL.

Look at the message, and make sure that all of the recipients' User Names are correct.

SECTION 17. GLOSSARY

ARPANET	Packet-switched network developed by the Defense Advanced Research Projects Agency.
BPS	Bits per second is the unit used for measuring line speed, the number of information units transmitted per second.
case dependent	Software differentiation between upper and lower case characters.
DDN	Defense Data Network, the DoD long-haul packet-switched computer communications network which includes the MILNET and the ARPANET as two of its subnetworks.
discard	To move a SAM message or form into the waste basket.
HERMES	Electronic mail program developed at BBNCC for DARPA.
host	Computer connected to a PSN on the DDN.
hostname	Name which officially identifies each computer attached to the DDN.
INFOMAIL	Electronic mail program developed by BBNCC.
keyboard macro	see MACRO.
LAN	Local Area Network; network of directly connected machines usually located within 10 miles of each other.
MH	Rand Corporation Mail Handling program for electronic mail.
macro	String of characters assigned to a SAM function key.
MILNET	The DDN unclassified operational military network.
MM	Electronic mail program developed at SRI International.
modem	Device which converts digital signals into analog signals (and back) for transmission over telephone lines (modulator and demodulator).
modem driver	Form which tells SAM the commands to use to set up and control a modem.
network service	SAM service that can be called by other services to avoid duplicating the same login script in several places.
NIC/Query	General information program on SRI-NIC.
NIC	DDN Network Information Center, located at SRI International, Menlo Park, CA.
NICNAME	See \WHOIS".
operating system	Software that supervises and controls tasks on a computer.
SAM Password	Password you select, which is needed to start SAM or use any SAM services.
shred	To delete a SAM message or form from the disk.
site	Organization or facility where a host is located.
service	SAM file that automates connection to a particular host.
SRI-NIC	DDN Network Information Center host located at SRI; the general information computer for the DDN.

SRI	SRI International, Menlo Park, CA, location of the DDN Network Information Center and early contributor to the development of the DDN.
TAC	Terminal Access Controller; special type of host attached to a PSN that allows direct terminal access to the DDN backbone.
TAC Access Code	Password assigned to DDN TAC users for TAC login.
TAC Userid	Alphanumeric character string that identifies a TAC user upon TAC login.
TACACS	TAC Access Control System; password system that limits use of TACs to authorized users.
TACNEWS	NIC program for reading DDN Newsletters and Bulletins and other items.
terminal	Communication device that lets a user send information to a computer by typing on a keyboard, and prints responses from the computer on paper or a screen.
TOPS20	Digital Equipment Corporation proprietary operating system which runs on the DEC 20 series of machines.
UNIX	An AT& T Bell Laboratories proprietary operating system which runs on large and small computers.
WHOIS	NIC program used to access the NIC electronic "white pages" database.

APPENDIX A. SAMPLE LOGIN SCRIPTS

In the sample scripts shown below, the actual script is in **bold** type, with comments on the right.

This script shows how SAM may be set up to require the user to enter their account name and password, useful if one SAM must be shared by several userswith differenct accounts on the same host..

do "DDN" .ha	Use the "Network: DDN" service to connect to the host.
wait for .hp	Wait for the variable ".hp", the host's prompt character.
prompt "Enter Account: " length 10	Prompt the user to type in the account name. The length of the account name is limited to 10 characters. SAM stores what the user types in in a temporary variable called "response".
send response CR	SAM sends the contents of the variable "response" (in this case the user's account name) to the host to begin the login process.
wait for "Password:"	SAM waits for the host to prompt for password.
prompt hide "Enter password: " length 10	SAM prompts the user to enter his password. The "hide" in this command causes the password to display only as X's on the screen. Length is again limited to 10 characters.
send response CR	SAM sends the contents of the variable "response" (now the user's password) to the host to begin the login process.
capture off	Turn off SAM's capture buffer.
clear buffer	Clear any characters in the buffer.

This login script asks the user what phone number to dial and tells SAM to use a different COM port that the default set in the PC Configuration basket. This can be useful if, for example, you have a direct line on one port and a modem on the other.

port com2	Set SAM to use COM Port 2.
prompt "Phone Number: "	Prompt the user to type in a phone number. SAM stores what the user types in in a temporary variable called "response".
dial '9,' response	Dial a '9' and then the number that the user entered at the prompt.
wait for 'CONNECT 1200'	Wait for a connect message from the host.
wait 1	Wait for line noise to clear.
send 'a'	Send a terminal type identifier.
wait for 'ase log in: '	Wait for host login prompt.
send .n1 CR	Send the contents of the .n1 field at the top of this service form, the account name, and a RETURN.

wait for CR LF 'KEY? '	Wait for host password prompt.
send .p1 CR	Send the contents of the .n1 field at the top of this service form, the account name, and a RETURN.

The following example shows an Access Procedure for a system that employs a call-back scheme. SAM dials the service, logs in, hangs up, waits for the call back, and answers the phone:

dial	Dials the phone using the default configuration.
wait 20 for CR	Wait 20 seconds to receive a RETURN from the remote system.
echo on	Turn keyboard echo on.
wait 20 for "Hello!"	Wait 20 seconds for "Hello" from the remote system.
wait 1	Wait 1 second.
send 'your id' CR	Send your system ID, then a RETURN.
echo off	Turn off keyboard echo.
hang up	Hang up the phone.
wait 300 for call	Wait 300 seconds for the remote system to call back.

APPENDIX B. HOW SAM USES THE PC

Keyboard	SAM uses standard BIOS calls.
Screen	SAM writes directly to the screen.
Memory	SAM has standard memory usage; 256k is require to run SAM, 320k to use the "push to DOS" feature.
Comm Ports	SAM configures only the comm port specified for communication, and talks directly to it. If a serial printer is selected in the Configuration form, SAM automatically assumes it will use the other comm port and configures it as specified.
Printer	SAM writes directly to the printer. To print a message with a redirected printer, copy the message to a file named "PRN", "LPT1", etc.
Mice	SAM is compatible with mice that use standard cursor-key emulation.
Game Port	Not used.

APPENDIX C. OBTAINING KERMIT

The Kermit File Transfer Protocol

Kermit is an error-correcting protocol for transferring sequential files between computers of all sizes over ordinary asynchronous telecommunication lines. Kermit is non-proprietary, thoroughly documented, and in wide use. The protocol and the original implementations were developed at Columbia University and have been shared with thousands of other institutions all over the world, many of which have made significant contributions of their own. Kermit is presently available for more than 200 different machines and operating systems, and additional versions are always under development.

All Kermit programs perform file transfer using the Kermit file transfer protocol. In addition, Kermit programs for personal computers also provide terminal emulation, usually of the DEC VT52, VT100, or similar terminal, and some of the mainframe Kermit programs are capable of initiating connections, acting as dumb terminals to remote systems. Kermit programs work only over asynchronous RS-232 direct or dialup connections, or connections that simulate them. For file transfer to take place, there must be a Kermit program running on *each* end of the connection, one on each computer.

There are Kermit programs for most popular "generic" operating systems, including UNIX, MS-DOS, and CP/M, and for mainframes and minicomputers from Burroughs, Cray, CDC, Data General, DEC, Gould (SEL), Harris, Honeywell, Hewlett-Packard, IBM, Perkin-Elmer, Prime, Sperry/Univac, and Tandem, and for particular microcomputers and workstations from Apple, Apollo, Atari, Commodore, IBM, Tandy, and many others, written in a wide variety of languages including many different assemblers, plus high-level languages like Algol, Basic, Bliss, C, Forth, Fortran, Lisp, Mumps, Pascal, PL/I, and Ratfor. A complete list of Kermit programs accompanies this flyer.

Here are some details about the several most popular Kermit programs. Most of the following implementations are capable of both local and remote operation, server and client modes, text and binary file transfer, and support a full range of communications options--speed, parity, duplex, flow control, handshake--to allow adaptation to a wide variety of hosts (including IBM mainframes) and communication media.

- **IBM PC Kermit** Version 2.29 runs under PC-DOS version 2.0 and later on the entire IBM PC family, as well as on IBM "clones" and compatibles. It provides nearly complete DEC VT102 terminal emulation at speeds up to 38.4K baud fully buffered and interrupt driven--and includes support for color displays, compatibility with various "desktop organizers," and selectable emulation of other terminals. There are also versions of Kermit specifically tailored for a variety of other MS-DOS systems, including the DEC Rainbow, Zenith-100, Victor 9000, HP-110/150, and many others, and there is a "generic" MS-DOS Kermit for systems not explicitly covered.
- **UNIX Kermit** is distributed only in C-language source form. It may be built for nearly any machine running practically any post-V6 variation of UNIX, including V7, Berkeley 2.x and 4.x, AT&T System III and System V, Xenix, Venix, and so on. The same source also serves as a basis for Macintosh, Amiga, and other Kermit programs.
- **VAX/VMS Kermit** is written in Bliss, but it is also distributed in Macro-32 and hex form, so that a Bliss compiler is not required. Other versions exist in C and Pascal.
- **IBM mainframe Kermit** programs for VM/CMS and MVS/TSO work only with asynchronous ASCII TTY connections through 3705 or equivalent front ends, or through Series/1, 7171, or similar protocol converters that support the Yale ASCII Communications System; beyond this exception, Kermit cannot be used to transfer files in the IBM 3270-style full-screen terminal environment. There are no Kermit programs for DOS/VSE, or IBM minis like the System/34 and System/38, because these systems do not support asynchronous ASCII communications. Currently, IBM mainframe Kermits run only in remote mode.

The Kermit software--including source code--is furnished free and without license, and without warranty of any kind, and neither Columbia University, nor the individual authors, nor any institution that has contributed Kermit material, acknowledge any liability for any claims arising from the use of Kermit. Furthermore, it must be stated that the quality of the Kermit programs varies--some are polished, well-documented professional products and others are not. Kermit programs are contributed by public-spirited volunteers, and Columbia University does not wish to discourage such contributions by subjecting them to a rating system. Since source code is provided for all implementations, users may make improvements or write documentation where it is lacking and are encouraged to contribute their work back to Columbia for further distribution. Under certain conditions (described in a separate document) software producers may include Kermit protocol in their products.

Although the Kermit software is free and unlicensed, Columbia University cannot afford to distribute it for free because the demand is too great. To defray our costs for media, printing, postage, labor, and computing resources, we require moderate distribution fees from those who request Kermit directly from us. The schedule is given on the accompanying Kermit Order Form. You may also obtain Kermit programs from many other sources, including user groups, networks, dialup bulletin boards, and you may copy them from friends, neighbors, and colleagues. In fact, you may obtain Kermit programs from anyone who is willing to share them with you, just as you may share them yourself.

Kermit is distributed by Columbia University primarily on 9-track magnetic tape, suitable for reading on most mainframe and minicomputers. It is assumed that Kermit will be ordered in this form by institutional computer centers, whose professional staff will take the responsibility for "bootstrapping" the microcomputer versions from the tape to diskettes for their users. The tapes include source code and any available documentation for each Kermit implementation, and in some cases also binaries (usually encoded in hex or other printable format). Selected microcomputer versions are also available from Columbia on diskette.

Documentation includes the *Kermit User Guide*, which contains complete instructions for using and installing the major implementations of Kermit, and the *Kermit Protocol Manual*, which is a guide to writing new Kermit programs. One printed copy of each manual is included with any tape order, and additional copies may be ordered separately. The manuscript from the Kermit article that appeared in the June and July 1984 issues of BYTE Magazine, and the book *Kermit, A File Transfer Protocol* (Frank da Cruz, Digital Press, 1986) may also be ordered separately.

Once you receive Kermit, you are encouraged to copy and redistribute it, with the following stipulations: Kermit should not be sold for profit; credit should be given where it is due; and new material should be sent back to Columbia University so that we can maintain a definitive and comprehensive set of Kermit implementations for further distribution. And finally, *please use Kermit only for peaceful and humane purposes.*

Ordering Information

There are two separate Kermit tapes, A and B. There are too many Kermit files to fit on a single tape (soon, there will be too many to fit on two tapes). All tapes are half-inch, 2400-foot, 9-track, 1600bpi, odd parity. They are available ONLY in the following formats:

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

Blocksizes, when applicable, are our choice and in the range 8K-10K (use of smaller blocksizes could overflow the tapes). NO OTHER FORMATS ARE AVAILABLE. We can NOT make 800bpi or

6250bpi tapes, unlabeled tapes (except as noted above), fixed-block tapes, or custom tapes of any kind. If none of the above formats looks familiar to you, then specify ANSI--this is an industry standard format that *should* be readable by any computer system.

Tapes include machine readable source for both programs and documentation.

TAPE "A" CONTAINS:

- The microcomputer (PC, workstation) Kermit implementations
- The Info-Kermit mail archive

TAPE "B" CONTAINS:

- The mainframe and minicomputer Kermit implementations.
- The Kermit User Guide and the Kermit Protocol Manual

EXCEPTIONS:

- C-Kermit is the basis of all Unix Kermit implementations, mainframe and micro. It is on tape B. Macintosh and Amiga Kermits are also generated from the C-Kermit sources, so they too are on tape B. Duplicate copies of the Macintosh and Amiga hex and doc files (no source) are also included on tape A for convenience.
- While the general documentation is on tape B, any documentation of a specific nature is distributed together with the program it describes.

Kermit diskettes may also be ordered in certain formats.

The following Kermit products are available:

1. **Kermit Distribution Tapes.** Specify Tape A (micros) or Tape B (mainframes), and tape format (ANSI, TAR, OS, CMS, DEC-10, or DEC-20). $100 PER TAPE.
2. **Kermit programs on diskette.** For IBM PC, XT, and AT; PC-DOS. $10 each.
3. **Printed Documents.**
   Book: *Kermit, A File Transfer Protocol* ($25);
   Kermit User Guide ($5 each)
   Kermit Protocol Manual ($5)
   BYTE Magazine article manuscript ($5)
4. **Shipping and Handling.** Outside North America, add $25; USA RUSH ORDERS (Sent Federal Express), add $20
5. **Sales Tax.** Do not add sales tax.

Make checks in U.S. Dollars, payable to:

COLUMBIA UNIVERSITY CENTER FOR COMPUTING ACTIVITIES

To order kermit, send your order (from the items listed above) with either a check for the total amount or a purchase order for that amount **plus a $100 billing fee** to:

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

North American orders are shipped by delivery service or first class US mail, and shipping costs are included. Overseas orders are shipped first class US mail; an additional shipping charge is required. Orders are normally processed within 2-4 weeks of receipt, but firm delivery schedules or methods cannot be guaranteed.

INDEX

SAM Warranty Registration Form

Please fill out this registration form and send to the DDN Network Information Center. Registration is required to be eligible for customer support.

Suggestions or comments about SAM may also be sent online to SAM@SRI-NIC.ARPA.

SAM User

Your name:

Organization:

Title/Rank:

Address:

City:

State:

Zip Code:

Country:

Phone Number:

Host Computer

Host Computer Name:

Mail Program:

Operating System:

Personal Computer

Computer Brand/Model:

Modem Brand/Model:

Location (work/home):

I learned about SAM from:

Other Comments:

fold

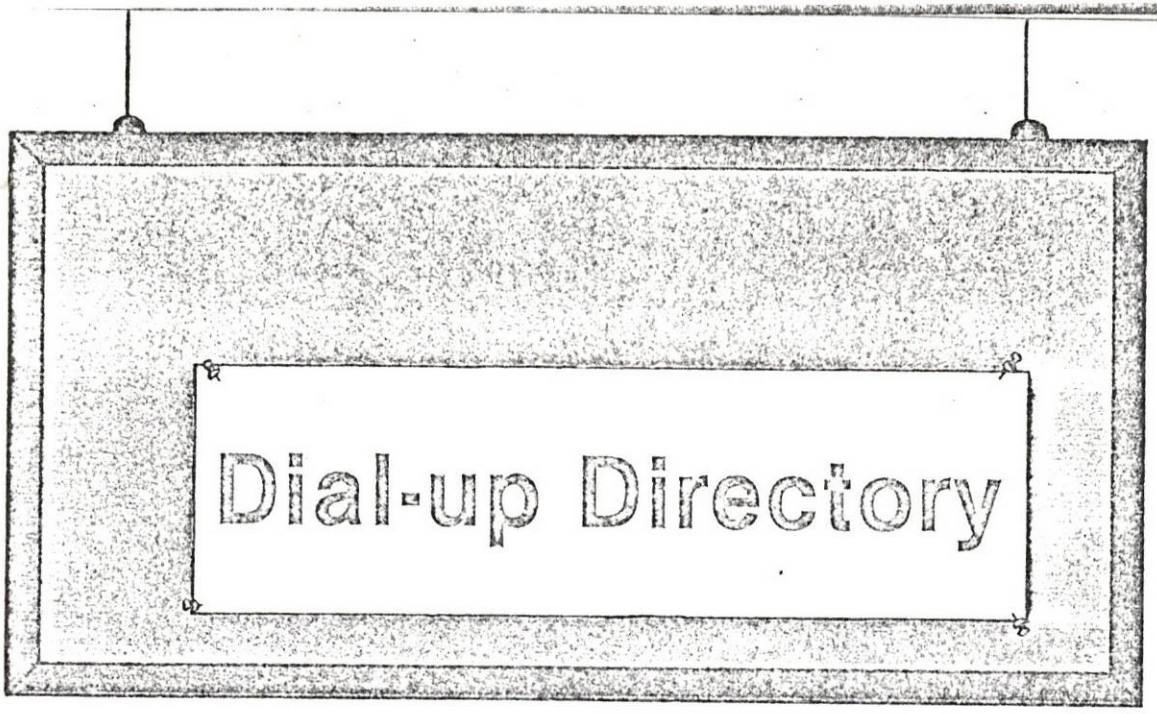
---

stamp

To: SRI International, Room EJ291
DDN Network Information Center
333 Ravenswood Avenue
Menlo Park, California 94025

---

fold



Computer bulletin board services are everywhere. To join the fun of instant information exchange, you'll need a terminal, a telephone and a modem (like the one described on p. 52).

Frank J. Derfler, Jr.
PO Box 17283
Montgomery AL 36117

The hallways of companies in the computer industry ring with phrases such as "distributed processing" and "smart terminals." Megadollar corporations are modifying their management structures to take advantage of the synergistic relationship between computers and communications. Flashy executives and congressmen too consider it a "perk" (nontaxable) to be able to dial into their mailboxes from a portable terminal and sort through their old and new messages. Military communications planners talk about many network terminals sharing a "data base in the sky." There is no reason why those of us with our own microcomputers can't participate in the exciting world of digital information transfer just like the megabuck boys.

Introduction

This is the start of a new *Microcomputing* feature called the Dial-up Directory. The Dial-up Directory will have two purposes: to provide (1) an annotated directory of those computer bulletin board services (CBBS) that exist around the country and of those individuals interested and capable of exchanging data by phone and (2) information on software and systems that can give you a dial-up capability.

We all have different interests and ways of utilizing our computers. Often our interests and requirements are not shared by local individuals or clubs. It is extremely helpful to be able to share programs and suggestions via data phone calls from around the country.

> We would like to publish the name and phone number of anyone presently interested in receiving data calls.

Whether your interests are graphics on the Apple, games on the PET, number crunching on the North Star or computer-assisted instruction on the TRS-80, there are others out there similarly inclined. We will try to hook you up.

We will have a lot of work to do together. We will have to work out and spread the word not only on electronic protocols, but also on those human protocols that exist whenever people interact with one another. We will describe ideal ways of doing things, the cheap way of doing things and the road down the middle. First, though, let's describe the world we will be looking at for those who may not be familiar with it.

Getting Started

Almost all of the computers we own have a practical communications capability of one sort or another. The cassette recorder port on most machines is one example.

The main I/O capability we are interested in is the RS-232 ASCII port available either stock or as an accessory on almost every microcomputer. Cassette and disk formats may differ between brands of computers and, indeed, even between models of the same brand, but the RS-232 ASCII port brings everything out in a common electrical medium of exchange. My OSI can talk to your TRS-80 at a useful speed, and we can exchange programs and information over a communications link.

Probably the best (but certainly not the only) communications medium we have between us is the telephone line. The U.S. still has the best overall phone system in the world (Japan and some sections of the Middle East are coming up fast), and the telephone represents an economical way of sending our minds out around the country.

In order to convert and send the digital plus and minus voltages of the RS-232 signal over the phone lines, we need a device called a modem, which converts these dc voltages into audio

From:	EDST	EST/ CDST	CST/ MSDT	MST/ PDST	PST	HST AST
To GMT Add: Hours	+4	+5	+6	+7	+8	+10

Table 1. Conversion from local to Coordinated Universal Time (GMT).

tones. The tones are received by a modem at the distant end and converted back into dc. The Bell system set the standards for low-speed (to 300 baud) modems; their Bell 103A standard is typically used. Under this standard, each party (one called the "originate" and one called the "answer") uses a different set of tones.

This means that if I wanted to call you to send you the nifty program that I just wrote to water my vegetable garden, we would first have to verbally agree on the speed (110 and 300 baud are the most common) and on which one of us would use the originate signaling tones and which one would use the answer tones. Then we would connect our modems to our phones and send data.

Obviously, one of us would have to have a modem capable of operating in the answer mode. This is important, because as you read modem ads you should look for the capability you need. Many modems are originate only. Many others advertise themselves as originate/answer but don't make it clear that the option requires extensive rewiring. "Switchable originate/answer" is the key phrase for complete flexibility.

Potential Difficulties

Establishing contact by phone probably only means you are over the hardware hurdle. Another favorite buzz phrase in large system procurement today is that hardware is easy . . . it's software that's difficult. Once you receive my data on your system, what can you do with it? With the right software, your system can save it on disk or tape to recall and use again at your convenience. We will talk about software to do that in future articles.

Without the right software, you can only print out the data you receive. But at least you have a hard copy to refer to. If your computer acts only as a "dumb terminal," then you can probably have a nice chat, but you may have only a few scribbled notes to remember it by.

Other difficulties may be thrown into our exchange of data if I am not free to get on the phone at the same time you are. There are two ways around this: an auto answer capability to allow access with the terminal unmanned (after all, what good is automation if you can't put yourself out of a job?) and a store and forward service.

These services exist in many places around the country. They are typically known under the generic name of computer bulletin board services (CBBS). I can dial into this service (actually, anybody's system with an automatic answer modem, the right program, sufficient memory and a large electric bill), select the bulletins I want to read and leave a copy of my rutabaga-watering program.

In that way, you and everybody else on the system can review my program at your convenience. This is practically the ideal information exchange. Would you like to take part? That is the goal of this series.

The Directory

We would like to publish the name (use a pseudonym if you like, but no CB call signs, please) and phone number of anyone presently capable of and interested in receiving data calls. We will need any specifics or limitations, such as baud rate, answer only, special control codes or carriage returns. We need to know when and on what days you will be interested in receiving calls. We will also have room for information on interests—stock market analysis, for example.

One of our biggest services can be getting people with similar interests in touch with each other—digitally. Because of the various time zones involved, I suggest we use Coordinated Universal Time (also known as GMT, Zulu or WWV time). A quick-reference GMT-to-local-time conversion chart is included in Table 1.

Remember: You may be getting calls from around the country, so it is only common courtesy to keep your 5-year-old from answering the phone during the times you specified, and it might be nice to not answer at all if you are not interested or able to transfer data on a specific day. A firm promise to return the call at another time is probably the least you owe someone who called you in good faith. An automatic audio answering device such as a Code-a-phone will allow recording up to 30 seconds of received data. We'll also discuss transferring data from the Code-a-phone to the computer in a later article.

In this introductory article, let me acquaint you with three excellent computer bulletin board services (see Directory). They represent a good starting point because they each contain extensive prompts and guides to make your telecommunications trials less terrifying. They are all available 24 hours a day, work either 110 or 300 baud and operate in the answer mode. They are free of any financial charge and don't need any passwords or codes, but that can all change if they are abused. The rules are just like those for a campground: keep it clean, don't leave any garbage behind and don't overstay your welcome, because others are waiting to use the facilities.

You can enter any of these systems by dialing the phone number, connecting your modem as soon as you hear the answering tones begin and sending at least three carriage returns. The host computer will read the carriage returns and reply at the proper speed. It is then that the fun begins.

Let me hear from you if you want to receive data calls or if you operate a CBBS. Send mail to PO Box 17283, Montgomery AL 36117, or leave a message on the Atlanta CBBS (404) 939-1520. ◘

Location	Operated by	Phone
Dallas	Ric Martin and Bill Kennedy	(214) 641-8759
Atlanta	Les Freed	(404) 939-1520
Oregon	Jim Willing and Bill Marx	(503) 646-5510

Dial-up Directory.

ISI/RR-84-141

August 1984

University of Southern California

Annette L. DeSchon

MCI Mail/ARPA Mail Forwarding

INFORMATION SCIENCES INSTITUTE

213/822-1511

4676 Admiralty Way/Marina del Rey/California 90292-6695

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MCI Mail/ARPA Mail Forwarding		5. TYPE OF REPORT & PERIOD COVERED Research Report 6. PERFORMING ORG. REPORT NUMBER ISI/RR-84-141
7. AUTHOR(s) Annette L. DeSchon		8. CONTRACT OR GRANT NUMBER(s) MDA903 81 C 0335
9. PERFORMING ORGANIZATION NAME AND ADDRESS USC/Information Sciences Institute 4676 Admiralty Way Marina del Rey, CA 90292-6695		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA 22209		12. REPORT DATE August 1984 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This document is approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

---

18. SUPPLEMENTARY NOTES

---

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ARPA mail system, ARPANET, computer mail, electronic mail, Intermail, mail forwarding, mail system interconnection, MCI mail

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This document provides a description of the programs that are used to forward mail between the ARPA mail system and the MCI mail system (MCI Mail). It assumes that the reader is familiar with the ARPA mail system.

ISI/RR-84-141

August 1984

University of Southern California

Annette L. DeSchon

MCI Mail/ARPA Mail Forwarding

INFORMATION SCIENCES INSTITUTE

213/822-1511

4676 Admiralty Way/Marina del Rey/California 90292-6695

This research is supported by the Defense Advanced Research Projects Agency under Contract No. MDA903 81 C 0335. Views and conclusions contained in this report are the author's and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. Government, or any person or agency connected with them.

ISI Reprint Series

This report is one in a series of reprints of articles and papers written by ISI research staff and published in professional journals and conference proceedings. For a complete list of ISI reports, write to

Document Distribution
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
USA

Table of Contents

MCI Mail / ARPA Mail Forwarding

Annette L. DeSchon

1. Introduction

This document provides a description of the programs that are used to forward mail between the ARPA mail system and the MCI mail system (MCI Mail). It is assumes that the reader is familiar with the ARPA mail system programs and procedures. Section 2 describes the use of the Intermail automatic mail forwarding system [1]. Instructions are also included on how to set up the source routing information needed to forward mail between the ARPA mail system and MCI Mail. Section 3 contains a detailed description of the message files that may be input to the MCIMAILER program. MCIMAILER is the program in the Intermail system that actually interfaces to the MCI mail system, using a dial-out modem. The path file and the account files which are used by the MCIMAILER program are also discussed. Section 4 is a brief summary of the guidelines that should be followed when composing a message which is to be forwarded to MCI Mail. Lastly, section 5 contains examples which illustrate the various types of messages that might be sent.

In the examples which appear in this document, some fields in a message (or in a file) consist of a generic description (i.e., "sender's-Internet-mailing-address"), while others are more literal (i.e., "Smith@USC-ISIF"). This is intended to document both the format of the message, and the format of each field in the message. It should be assumed that each line of text in an example ends with a "carriage return" and a "line feed".

For historical reasons, an ARPA mail system mailbox will be referred to as an "ARPANET mailbox". It should be noted these mailboxes may be located on any host which is a part of the Internet. The Internet mailing addesses mentioned in this document appear in the form of ARPANET mailing addresses (i.e. "Smith@USC-ISIF" is assumed to be the short form of "Smith@USC-ISIF.ARPA"). A full Internet mailing address in the form "USER@HOST.DOMAIN" may be used anywhere that the short form is used.

The background information on MCI Mail is from two MCI Mail Customer Service publications: the *MCI Mail Service Guide* [4] and the *Basic User's Guide* [3]. Additional information is available through the on-line MCI Mail "HELP" commands. *Gateways, Bridges, and Tunnels in Computer Mail* by Danny Cohen and Jon Postel [2] is recommended for more information on the theory behind the interconnection of computer mail systems.

2. Mail Forwarding

The Intermail system currently consists of five programs which run on USC–ISIF, a DEC–SYSTEM TOPS–20 system. The IM program examines the source routing section of each message that enters the system, and formats it for processing by the appropriate mail interface program. The four mail interface programs are MM, MAILER, TELEMAILER, and MCIMAILER. The MM program is used to move mail out of the Intermail mailbox into separate files. The MAILER program picks up the mail which is destined for an ARPANET mailbox. The TELEMAILER program exchanges messages with the TeleNet mail system (Telemail) using a permanent leased line. The MCIMAILER program exchanges messages with MCI Mail using a dial–out modem.

Figure 2–1 illustrates the interaction of the various programs that comprise the Intermail system. A batch job which logs into the <INTERMAIL> directory is used to run these programs in the following order:

- MM reads messages from Intermail's ARPANET mailbox into "IN1.IM" files.
- IM converts "IN1.IM" files to "(--UNSENT-TELEMAIL--).IM", "(--UNSENT-MCIMAIL--).IM", or "[--UNSENT-MAIL--].USER@HOST" files.
- TELEMAILER sends messages from "(--UNSENT-TELEMAIL--).IM" files to Telemail and reads messages from the Telemail "Intermail" account into "IN1.IM" files.
- MCIMAILER sends messages from "(--UNSENT-MCIMAIL--).IM" files to MCI Mail and reads messages from the MCI Mail "Intermail" account into "IN1.IM" files.
- IM converts "IN1.IM" files to "(--UNSENT-TELEMAIL--).IM", "(--UNSENT-MCIMAIL--).IM", or "[--UNSENT-MAIL--].USER@HOST" files.
- MAILER picks up "[--UNSENT-MAIL--].USER@HOST" files to be delivered to ARPANET mailboxes.

The login information for MCI Mail and Telemail comes from the ".PTH" files. The "MAIL.TXT" files are normal TOPS–20 mailboxes which may have messages appended to them, or messages read from them. The other files shown contain messages (one per file) at various stages in the forwarding process.

2.1. Forwarding Mail from ARPA Mail to MCI Mail

Automatic mail forwarding may be used by anyone with an ARPANET mailbox. The message to be forwarded is created using one of the standard mail programs (e.g., MSG, MM, HERMES). The user then sends the message as ARPA mail, to "Intermail@USC–ISIF". It is then read from the Intermail

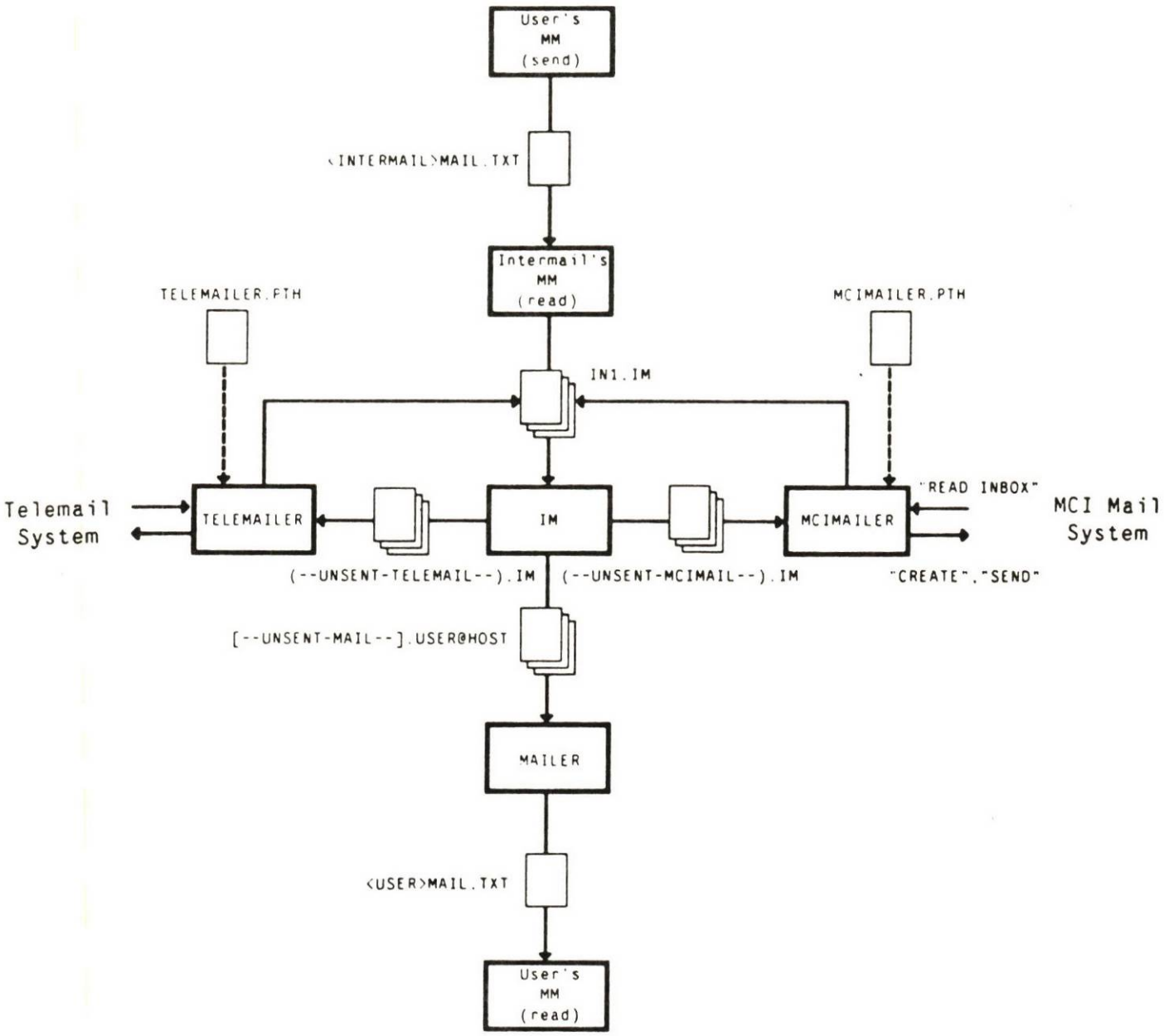


Figure 2-1: Intermail Programs

mailbox and processed by the IM mail forwarding program. The MCIMAILER program then dials out to the MCI mail system and sends any messages which are waiting in the <INTERMAIL> directory to be forwarded to MCI Mail. Next, it reads any MCI mail which is waiting in the Intermail MCI Mail "INBOX" into TOPS-20 files in the <INTERMAIL> directory. The IM mail forwarding program then reformats these files as ARPA mail and sends each message to the appropriate Internet mailing address. Using this method, any authorized user can send MCI Mail from a host which supports ARPA mail, using his usual mail program, and receive MCI mail in his ARPANET mailbox.

When mail forwarding is being used, the necessary addressing information is included by the user creating the message, at the beginning of the ARPA mail text field. This "source routing section" is separated from the rest of the text by a blank line. The form of the message as seen by the IM program is as follows:

```
ARPA Mail Header Section
<blankline>
ARPA Mail Text Section:
  Source Routing  Section
  <blank line>
  Message Text
```

The source routing section will contain several FSR (Forward Source Route) lines, which specify the path that the message will take as it is being forwarded. The first FSR is the Internet mailing address of the mail forwarding program. The second FSR is used by the IM program to create the address section of the message for the next hop that the message will take. For the purposes of forwarding ARPA mail to MCI Mail, this second FSR will be used to create the header fields of the MCIMAILER message file. In addition, the source routing section must contain at least one RSR (Return Source Route) line which may ultimately be used by the end recipient to make up the source routing section of a reply. Each FSR or RSR consists of a network or mail system name and the name of a user as he is known in the destination network. The network of the first FSR must match the network of the first RSR. This is an example of the source routing section in the simplest case:

```
FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]recipient's-MCI-address
RSR: [ARPANET]sender's-Internet-mailing-address
```

In forwarding mail to MCI Mail, an account number may be required. Account numbers are assigned by the person responsible for the MCI Mail account that is being used; for a USC-ISIF Intermail account number, see Jon Postel. The account number should appear on a separate line, following the appropriate FSR. When an account number is to be used, the source routing section consists of:

```
FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]recipient's-MCI-address
Account: account-number
RSR: [ARPANET]sender's-Internet-mailing-address
```

A multi-line format FSR or RSR may be used when the MCIMAILER header field information does not fit onto a single line. This format must be used when a postal address is needed, or when the list of recipients is too long for one line. The multi-line format also allows the creator of the message to specify optional MCIMAILER fields such as the "Cc" field, the "Subject" field, and the "Handling" options. In a multi-line format FSR or RSR, the first line contains only the label and the network or mail system name. The lines following, up to the next FSR or RSR, will be copied into the MCIMAILER message file (e.g., "(--unsent-MCIMAIL--).IM"). Therefore, the guidelines in section 3.4 define the format of the address section of a multi-line FSR or RSR, as well as the format of an MCIMAILER message file. A source routing section containing a multi-line format FSR with an account number might look like:

```
FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]
To: recipient's-MCI-ID
Cc: someone-else's-MCI-ID
Account: account-number
RSR: [ARPANET]sender's-Internet-mailing-address
```

As a message is forwarded, the source routing section is modified by the IM program to reflect each hop that the message has taken. Therefore, the source routing section of the message as it is received by an MCI Mail user is not the same as it was when it was originally sent. In addition, the header section (or "envelope") that the MCI Mail recipient sees is the header section that was created when the message was sent to the MCI Mail system by the MCIMAILER program. For example, the following ARPA mail:

```
Date: 18-May-84 11:52:21-PST
To: Intermail@USC-ISIF
From: DeSchon@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]123-4567
Account: 1234
RSR: [ARPANET]DeSchon@USC-ISIF

This the text of the message.
```

might be delivered to the MCI Mail recipient as:

```
Date:      Fri May 18, 1984 11:58 am  PST
From:      Intermail / MCI ID: 107-8239

TO:      * John Smith / MCI ID: 123-4567
Subject:   Sample Message

FSR: [MCIMAIL]123-4567
RSR: [MCIMAIL]Intermail
RSR: [ARPANET]DeSchon@USC-ISIF

This the text of the message.
```

2.2. Forwarding Mail from MCI Mail to ARPA Mail

Users of MCI Mail may also send mail to users having an ARPANET mailbox. To accomplish this, the MCI Mail user uses the MCI Mail "CREATE" and "SEND" commands to send mail to the MCI Mail "Intermail" (MCI-ID 107-8239) account. Later the MCIMAILER program, which is being run on a TOPS-20 system, dials MCI Mail and reads the messages from the Intermail "INBOX" into TOPS-20 files. The IM mail forwarding program then picks up these files, converts them to the proper format, and sends them as ARPA mail to the user(s) named in the source routing section of the message. As in messages being forwarded from the ARPA mail system to MCI Mail, the text of the message should include a source routing section, separated from the rest of the text by a blank line. The "Subject" field in the ARPA mail will be copied from the "Subject" field of the MCI Mail message. Any other MCI Mail header fields will be ignored. The following example illustrates the format of such a message:

```
To:      Intermail
Subject: Sample Message
Text:

FSR: [MCIMAIL]Intermail
FSR: [ARPANET]recipient's-Internet-mailing-address
RSR: [MCIMAIL]sender's-MCI-ID

This is the text of the message.
```

Or more specifically:

```
To:      Intermail
Subject: Sample Message
Text:

FSR: [MCIMAIL]Intermail
FSR: [ARPANET]Cohen@USC-ISIB
RSR: [MCIMAIL]105-0002

This is the text of the message.
```

The following procedure may be used to construct the source routing section of a reply to a message that has been forwarded through Intermail. Starting with the message as it is received, exchange the FSRs for RSRs, and the RSRs for FSRs. Then place the FSRs in front of the RSRs, at the beginning of the source routing section. For example, if a message from Intermail contains the following source routing section:

```
FSR: [MCIMAIL]123-4567
RSR: [MCIMAIL]Intermail
RSR: [ARPANET]DeSchon@USC-ISIF
```

the reply would contain the following:

```
FSR: [MCIMAIL]Intermail
FSR: [ARPANET]DeSchon@USC-ISIF
RSR: [MCIMAIL]123-4567
```

3. The MCIMAILER Program

The MCIMAILER program may be used directly by any TOPS–20 user who has his own MCI account. It runs on TOPS–20 and may be started at the EXEC level or from a batch job. Using account information read from a file ("MCIMAILER.PTH") in the connected directory, MCIMAILER automatically logs onto MCI Mail, reads mail, sends mail, and logs off. Messages read from MCI Mail are stored as ASCII files, which can be automatically delivered to the user's ARPANET mailbox by a TOPS–20 mail program, MAILER. Messages to be sent to an MCI Mail user can be created using either a mail program or a text editor. When the MCIMAILER program has completed the processing on a message destined for MCI Mail, files containing messages that have been sent will be renamed "MCI.SENT". Files containing messages that were not sent due to syntax errors in the message, or possibly line noise, will be renamed "MCI.NOT-SENT". Figure 3–1 illustrates the use of the MCIMAILER program by an ARPA mail user who has his own MCI Mail account.

3.1. Batch Control File

In order to receive mail in a timely manner, the user may want to submit a batch job which runs the MCIMAILER program and reschedules itself to run again at a later time. For example, the following commands might be placed in a file named "MCI.CTL":

```
@submit mci/batch-log:supersede/after:+04:00:00/output:nolog
@
@<INTERMAIL>MCIMAILER
@
```

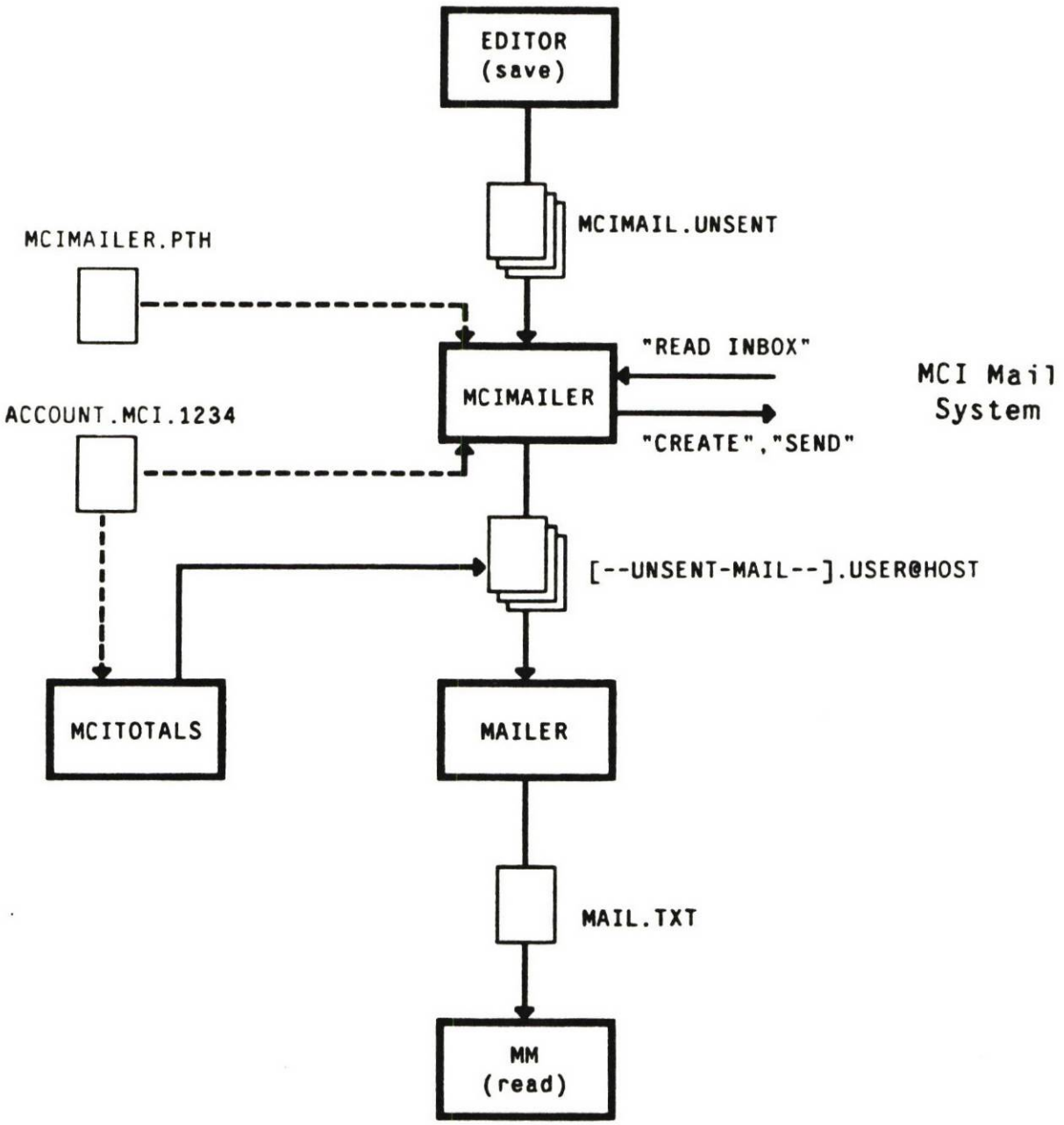


Figure 3-1: MCIMailer Program

This batch job may be submitted using the TOPS-20 command "SUBMIT MCI.CTL". Each time that the job runs it will resubmit itself to run again in four hours (creating a new log file each time), and then run the MCIMAILER program. When this method of running MCIMAILER is used, the "generation retention count" for the files "MCI.SENT" and "MCI.LOG" should be set using the TOPS-20 "SET FILE GENERATION-RETENTION-COUNT" command, to avoid a buildup of log files.

3.2. Path File

MCI account information is read from a file called "MCIMAILER.PTH" in the connected directory. Since this file contains a password, it should be given the "770000" file protection. The format is as follows:

```
number of the TTY line for the dial-out modem
phone number of MCI Mail
MCI Mail login name
MCI Mail password
mail-read-in file name (MCI Mail -> TOPS-20 direction)
mail-to-be-sent file name (TOPS-20 -> MCI Mail direction)
ACCOUNT REQUIRED (optional)
```

In addition to the information required to log on to an MCI Mail account, the input and output file names are specified. To have the mail that is read from MCI Mail automatically picked up by a mail program and delivered to an ARPANET mailbox, the "mail-read-in file name" should be:

```
<CTRL-V>[--UNSENT-MAIL--<CTRL-V>].username<CTRL-V>@hostname
```

NOTE: On TOPS-20, "control V" is used to quote the special characters in a file name. The files which contain messages to be sent via MCI Mail may be created using a text editor or a mail program. The "mail-to-be-sent file name" can be anything that is convenient to the user; Intermail itself uses "(--UNSENT-MCIMAIL--).IM".

The "ACCOUNT REQUIRED" line is optional. When it is included in an MCIMAILER path file, it specifies that the MCIMAILER program using the path file is to check each message that is being sent for a valid account number. If the "ACCOUNT REQUIRED" line does not appear, messages containing no account field will be accepted. If, however, an account field is included in a message, it must be a valid one.

An example of a possible MCIMAILER.PTH file is as follows:

```
64
6201449
JSmith
password
↑V[--UNSENT-MAIL--↑V].Smith↑V@ISIF
UNSENT-MCIMAIL.TXT
ACCOUNT REQUIRED
```

3.3. Account File

When the MCIMAILER program processes the account field of a message, it looks for a file named "ACCOUNT.MCI" with a version number corresponding to the account number in the message. This file must exist either in the connected directory, or in the login directory for the job running MCIMAILER. For example, when the message contains the MCIMAILER field "Account: 1234", the file "ACCOUNT.MCI.1234" must exist in order for the message to be sent. Each account file contains the following:

```
comment line
results file name
4HOUR count
ONITE count
PAPER count
REGULAR count
```

The account file is made up of ASCII characters and may be created using a text editor. Initially, it might look like:

```
Smith MCI Mail Account
↑V[--unsent-mail--↑V].Smith↑V@ISIF
0
0
0
0
```

In this example, as messages which use account "1234" are sent, the MCIMAILER program updates the counts in "ACCOUNT.MCI.1234" (See figure 3-1). When the MCITOTALS program is run, the totals will be listed in the results file, which in this example will be delivered to the ARPANET mailbox, "Smith@USC-ISIF". Each time that the MCITOTALS program is run, and a results file is produced, the counts in the account file will be reinitialized to zero, and the creation date of the file will be updated to reflect the current date. The MCIMONTHLY program performs the same functions as the MCITOTALS program except that it operates only on account files with a creation date in a different month from the current month.

3.4. MCIMAILER Fields

The fields that are recognized by the MCIMAILER program are:

- To,
- Cc,
- MCIMail-To,
- MCIMail-Cc,
- Subject,
- Account, and
- Handling.

All other fields will be ignored.

The "Account" field is used by the MCIMAILER program to keep track of the number of messages sent. This will enable the person responsible for an MCI Mail account to verify that the billing is correct. The account field may appear in any position relative to the other fields.

The "To", the "MCIMail-To", the "Cc", and the "MCIMail-Cc" fields will be referred to as "name" fields. A name field can take several forms. For purposes of this discussion, an MCI-ID is defined as a unique MCI login name, the name of a mailing list that has been previously created, or a unique seven digit MCI Mail identification code. A "name" is defined as an MCI-ID followed by an option list. Using these definitions, a name field can take the following forms:

- A prompt followed by a name.

```
To: Postel (PAPER)
```

- A prompt followed by a list of names.

```
To: Postel, 123-4567 (ONITE)
```

- A prompt followed by a name (including an option list containing the "ADDRESS" option), followed by up to four lines of indented postal address.

```
To: Jon Postel (ADDRESS, ONITE)
    USC Information Sciences Institute
    4676 Admiralty Way
    Marina del Rey, CA  90292-6695
```

- A prompt followed by text (which is ignored), followed by "(Attn:" name ")". In this example, "Smith@USC-ISIF" is treated as a comment, and the message is sent to "Postel".

```
To: Smith@USC-ISIF (Attn: Postel (ONITE,PAPER))
```

The "MCIMail-" prefix (on the "MCIMail-To" prompt or the "MCIMail-Cc" prompt) causes the program to ignore any names appearing previously in the "To" field or the "Cc" field, respectively. Thus, a message consisting of:

```
To: Smith@USC-ISIF
Cc: Jones@USC-ISIF
Sender: Ross@USC-ISIF
MCIMail-To: Postel (PAPER)
MCIMail-Cc: 123-4567
Subject: Sample Message

This is the text of a sample message.
```

would be be input to MCI Mail by the MCIMailer program as:

```
To: Postel (PAPER)
Cc: 123-4567
Subject: Sample Message

This is the text of a sample message.
```

If multiple "To" (or "Cc") fields appear, they will all be used. Multiple "Subject" fields are unlikely, but if they should appear, only the last will be used. Some messages that constitute legal and reasonable input to the MCIMAILER program may not be acceptable as input to some of the standard ARPA mail programs. If this is the case, a text editor may be used to create or add to a message. Multiple fields and postal addresses probably fall into this class. For example, a message consisting of

```
To: Postel (PAPER)
Cc: 123-4567, Smith, Jones
Cc: Annette DeSchon (ADDRESS)
    USC Information Sciences Institute
    4676 Admiralty Way
    Marina del Rey, CA  90292-6695
Subject: Sample Message

This is the text of a sample message.
```

would be be input to MCI Mail as:

```
To: Postel (PAPER)
Cc: 123-4567
Cc: Smith
Cc: Jones
Cc: Annette DeSchon (ADDRESS)
    USC Information Sciences Institute
    4676 Admiralty Way
    Marina del Rey, CA  90292-6695
Subject: Sample Message

This is the text of a sample message.
```

The "Handling" field is used to specify that a particular letterhead or a particular signature be used on a printed letter. To use these features, the MCI Mail account which is being used must subscribe to the "advanced user" service. In addition, the letterhead and the signature must registered with MCI Mail in advance. All the regular handling options described in the next section can be specified in the option list which follows the name of each addressee in the MCIMAILER name field.

3.5. MCI Mail Handling options

The MCI Mail handling options as described in the MCI Mail documentation [4] are as follows:

- Electronic delivery, the "instant letter", is the default. ($1)
- The "PAPER" option is used to specify that a paper copy of the message, or "MCI letter", be mailed to an MCI Mail subscriber at the postal address on record in his account. ($2)
- The "ADDRESS" option is the same as the "PAPER" option except that the postal address is entered explicitly. This means that the recipient does not have to be an MCI Mail suscriber. ($2)
- The "ONITE" option specifies that the message be delivered overnight by a courier if a paper copy has been specified. A message posted by 8 pm Eastern time Monday through Friday will be printed and hand delivered by noon the next business day. If it is posted from 8 pm Friday night through Sunday midnight, it will be delivered on Monday morning. Electronic copies will be marked "**PRIORITY". ($6)
- The "4HOUR" option specifies that a message posted between the hours of 6 AM and 6 PM, in the addressee's time zone, Monday through Friday, be printed and hand delivered within 4 hours. Letters posted after 6 PM will be delivered by 10 AM the next business day. Electronic copies will be marked "**PRIORITY". ($25)
- The "RECEIPT" option causes a delivery acknowlegement to be sent back to the sender. The time that the acknowlegement occurs depends on the send option. For electronic delivery the acknowlegement occurs when the recipient reads the message. For regular paper delivery it occurs when the letter is placed in the mail. For overnight delivery it occurs when the letter has been given to a courier. For 4 hour delivery it occurs when the courier has delivered the letter. (No Charge)

Laser-printed page dimensions are 80 characters wide and 54 lines long. There may be fewer lines on the first page to accomodate the letterhead. For 4-hour service and overnight service, it is advisable to confirm that MCI Mail delivers to the addressee's zip code area. This information may be found in the *MCI Mail Service Guide* [4] or online on the MCI mail system through use of the "HELP" command, specifying the state in which the letter is to be delivered.

For delivery by a courier, the 4HOUR or ONITE letter must have:

- A street address, not a post office box.
- The correct zip code at the end of the last address line.
- Someone at the destination to receive the letter.

Mistakes are the sender's responsibility! The sender will be charged for the message even when the address or the zip code is entered incorrectly, making the message impossible to deliver. MCI Mail suggests including the phone number of a person who can be contacted in the event of a delivery problem. This phone number should appear between the name of the recipient and the rest of the address.

4. Summary of MCI Mail Guidelines

The following guidelines apply to mail being sent to Intermail for forwarding to MCI Mail:

1. The text must begin with a source routing section, followed by a blank line. The network of the first FSR and the first RSR must match.
2. A multi-line format FSR should be used in the following cases:
   - The recipients' addresses will not fit on one line.
   - A postal address is used.
   - A "Cc" field or a "Handling" field is used.
3. The subject of the message (if any) may be in the "Subject" field of the ARPA mail header section of the message, or in a "Subject" field as part of a multi-line format FSR.
4. An account number may be required.
5. The "RECEIPT" option will not work when the message is forwarded through "Intermail". This is because the acknowlegement that is generated by MCI Mail will not contain the source routing section needed by the IM program to send the acknowlegement back to the sender's ARPANET mailbox.

The following guidelines apply to both messages that are being forwarded and messages that are used as direct input to the MCIMAILER program:

6. MCIMAILER field names (or the field names in a multi-line format FSR/RSR), must start at the beginning of a line. Postal addresses must be indented. If a postal address is to be used, the list of MCI Mail handling options following the recipient's name must include the "ADDRESS" option. Unless a postal address is being used, a field may not consist of more than one line.

7. No parentheses may appear in the first line of a name field unless they are part of the handling options list.

8. The maximum length for a line which will be used in the MCI Mail header (not including the prompt or the leading blanks) is 40 characters. The maximum number of lines allowed for a name and postal address is five. Any additional lines will be ignored.

9. At least one name in the "To" field is required. All other fields are optional.

10. The last line of text must be followed by a carriage return.

11. Text may not contain a "/" on a line by itself.

12. When a message is being read from MCI Mail to TOPS-20 by the MCIMAILER program, any line of text that matches an anticipated prompt will be treated as a real prompt.

5. Examples

5.1. Forwarding Mail from ARPA Mail to MCI Mail

This section contains examples of messages which could be forwarded from the ARPA mail system to MCI Mail. They can be created using either a mail program or a text editor. Note that because they are being forwarded, they are all addressed to "Intermail@USC-ISIF". The address of the MCI Mail user who is to receive the message is included in a source routing section at the beginning of the text. It is important to include the name of the person whom the message is from in the "RSR" (and perhaps elsewhere in the text) since the message will be from the MCI user "Intermail" when it is delivered on the MCI Mail system.

In this simple example, DeSchon (an ARPA mail user) is sending a message to MCI Mail user John Smith (MCI-ID 123-4567) through the Intermail system.

```
To: Intermail@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]123-4567
RSR: [ARPANET]DeSchon@USC-ISIF

This is the text of the sample message.

-- Annette
```

Assuming that John Smith's MCI Mail user name (JSmith) is known to be unique, the following message is equivalent:

```
To: Intermail@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]JSmith
RSR: [ARPANET]DeSchon@USC-ISIF

This is the text of the sample message.

-- Annette
```

Using a postal address:

```
To: Intermail@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]
To: John Smith (ADDRESS)
    1234 Maple Street
    Los Angeles, CA  90049
RSR: [ARPANET]DeSchon@USC-ISIF

This is the text of the sample message.

-- Annette
```

Along with an additional (electronic mail) addressee, a paper copy, and an "Account" field:

```
To: Intermail@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]
To: Postel
To: John Smith (ADDRESS)
    1234 Maple Street
    Los Angeles, CA  90049
Cc: ADeSchon (PAPER)
Account: 1234
RSR: [ARPANET]DeSchon@USC-ISIF

This is the text of the sample message.

-- Annette
```

In following example of an urgent message, the phone number of the person who should be contacted if there are delivery problems is included between the name of the addressee and the first line of the address. Note that only John Smith's copy of the message need cost $25. The assumption is that the regular "instant letter" will reach Postel quickly enough, and that a regular "MCI letter" sent through the U.S. mail will be adequate for ADeSchon's copy.

```
To: Intermail@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]
To: Postel
To: John Smith (ADDRESS,4HOUR)
    (213)822-1511 x159
    1234 Maple Street
    Los Angeles, CA  90049
Cc: ADeSchon (PAPER)
Account: 1234
RSR: [ARPANET]DeSchon@USC-ISIF

This is the text of the sample message.

-- Annette
```

5.2. Forwarding Mail from MCI Mail to ARPA Mail

This section contains examples of messages which could be forwarded from MCI Mail to the ARPA mail system. Note that because they are being forwarded, they are all addressed to the MCI user "Intermail". The Internet mailing address of the recipient of the message is included in a source routing section at the beginning of the text. It is important to include the name of the person whom

the message is from in the "RSR" (and perhaps elsewhere in the text) since the message will be from "Intermail@USC-ISIF" when it is delivered to the recipient's ARPANET mailbox.

In this example, MCI Mail user JSmith (MCI-ID 123-4567) is sending a message to DeSchon (an ARPA mail user) through the Intermail system.

```
To: Intermail
Subject: Sample Message

FSR: [MCIMAIL]Intermail
FSR: [ARPANET]DeSchon@USC-ISIF
RSR: [MCIMAIL]123-4567

This is the text of the sample message.

-- John
```

5.3. Input to the MCIMAILER Program

This section contains examples of messages which could be used as direct input to the MCIMAILER program. Note that when the message is delivered on the MCI Mail system, it will be "From" the MCI Mail login name specified in the MCIMAILER.PTH file that the MCIMAILER program is using.

This is an example of a message addressed to John Smith, which could be created using a text editor.

```
To: 123-4567
Subject: Sample Message

This is the text of the sample message.
```

When a mail program is used to create a such a message, it may be necessary to use an "Attn" or an "MCIMail-To" field to avoid having the adressee's MCI Mail user name disallowed by the mail program. Since the mail program will not accept "123-4567" as a valid Internet mailing address, DeSchon used herself for the part of the address that is checked by the mail program, and included MCI-ID 123-4567 in an "Attn" field, which is not checked by the mail program.

```
To: DeSchon@USC-ISIF (Attn: 123-4567)
Subject: Sample Message

This is the text of the sample message.
```

Another way to accomplish the same thing is:

```
To: DeSchon@USC-ISIF
MCIMail-To: 123-4567
Subject: Sample Message

This is the text of the sample message.
```

In this example, a postal address is used. (There is probably no way to create this message using a mail program.)

```
To: John Smith (ADDRESS)
    1234 Maple Street
    Los Angeles, CA  90049
Subject: Sample Message

This is the text of the sample message.
```

Along with an additional (electronic mail) addressee, a paper copy, and an "Account" field:

```
Account: 1234
To: Postel
To: John Smith (ADDRESS)
    1234 Maple Street
    Los Angeles, CA  90049
Cc: ADeSchon (PAPER)
Subject: Sample Message

This is the text of the sample message.

-- Annette
```

The following is an example of the use of the 4HOUR handling option:

```
Account: 1234
To: Postel
To: John Smith (ADDRESS,4HOUR)
    (213)822-1511 x159
    1234 Maple Street
    Los Angeles, CA  90049
Cc: ADeSchon (PAPER)
Subject: Sample Message

This is the text of the sample message.

-- Annette
```

5.4. Sending MCI Mail to an ARPA Mail User with an MCI Mailbox

There are no special guidelines for a message which is sent by one MCI Mail user to another MCI Mail user and read using the MCIMAILER program. Since all of the messages in the recipient's MCI Mail "INBOX" are for the recipient himself, no additional addressing information is required. For example,

```
To: ADeSchon
Subject: Sample Message

This is the text of the sample message.

-- John
```

.....

References

[1] D. Cohen.
A Suggestion for Internet Message Forwarding for MOSIS.
IEN 180, USC Information Sciences Institute, March, 1981.

[2] D. Cohen and J. Postel.
Gateways, Bridges, and Tunnels in Computer Mail.
Technical Report ISI/RS-83-117, USC Information Sciences Institute, January, 1984.
Also appears in Local Networks: Distributed Office & Factory Systems, Proceedings of Localnet '83, New York, and Local Networks: Strategy and Systems, Proceedings of Localnet '83, New York.

[3] Basic User's Guide
MCI Mail, 2000 M Street, N.W., Washington D.C. 20036, .

[4] MCI Mail Service Guide
MCI Mail Customer Service, Box 1001, 1900 M Street, N.W., Washington D.C. 20036, .

(0) NACC: 2016-84
(1) DOC-TYPE: 4
(3) REPORT-NO: ISI/RR-84-141
ISI RR 84 141
(11) CONTRACT-NO: MDA-903-81-C-0335
MDA 903 81 C 0335
(16) PERSONAL-AUTHOR: DeSchon, A.L.
(17) AUTHOR-AFFIL: University of Southern California, Information Sciences Inst
(22) TITLE: MCI mail/ARPA mail forwarding
MCI mail ARPA mail forwarding
(23) PLACE-PUB: Marina del Rey, CA
(24) PUBLISHER: University of Southern California, Information Sciences Inst.
(26) PUB-DATE-MR: 840800
(27) PUB-DATE: 1984 August
(28) PAGES-TOTAL: 21 p
(67) MEDIA-TYPE: p
(70) PRIM-SEC-REF: 2
(74) KEYWORDS: ARPA
Advanced Research Projects Agency
ARPA mail system
Computer mail
Electronic mail
Intermail
Mail forwarding
Mail system
MCI mail
MCI ARPA mail interconnection
Mail routing
Message routing
Message header
Message file
FSR
RSR
Forward Source Route
Return Source Route
(75) ABSTRACT: The programs used to forward mail between the ARPA (Advanced Research Projects Agency) mail system and the MCI mail system are described in this paper. This forwarding is possible through the Intermail automatic mail forwarding system, which is described. Directions are also given for including routing information in a message. The MCIMAILER program is discussed, and the types of message files it will accept are detailed. The guidelines for forwarding mail to the MCI system are summarized, and examples provided of different types of messages.
(76) ENTRY-DATE: 860310
(77) UPDATE: 871209-145633 APRIL
(83) CPLACE: Marina del Rey, CA
(85) LOCATION: Main
(86) BIBSET: General Net
(88) CORP-ORG: University of Southern California, Information Sciences Inst.

ISI Research Report
RR-85-158
September 1985

University of Southern California

Annette L. DeSchon

INTERMAIL, An Experimental Mail Forwarding System

DDN Network Information Center
SRI International
Menlo Park, California 94025

INFORMATION SCIENCES INSTITUTE

213/822-1511
4676 Admiralty Way/Marina del Rey/California 90292-6695

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ISI/RR-85-158	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* INTERMAIL, An Experimental Mail Forwarding System		5. TYPE OF REPORT & PERIOD COVERED Research Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR*(s)* Annette L. DeSchon		8. CONTRACT OR GRANT NUMBER*(s)* MDA903 81 C 0335
9. PERFORMING ORGANIZATION NAME AND ADDRESS USC/Information Sciences Institute 4676 Admiralty Way Marina del Rey, CA 90292–6695		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA 22209		12. REPORT DATE September, 1985 13. NUMBER OF PAGES 18
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

This document is approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

.....

18. SUPPLEMENTARY NOTES

.....

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

ARPA mail system, ARPANET, COMPMAIL, computer mail, Dialcom, electronic mail, Intermail, mail forwarding, mail system interconnection, MCI Mail, Telemail

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

The INTERMAIL mail forwarding system connects the DARPA Internet mail system to a number of commercial mail systems, including MCI Mail, Telemail, and the IEEE CompMail system. It is implemented on a DEC TOPS-20 system using the SAIL programming language.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

ISI Research Report
RR-85-158
September 1985

University of Southern California

Annette L. DeSchon

INTERMAIL, An Experimental Mail Forwarding System

INFORMATION SCIENCES INSTITUTE

213/822-1511

4676 Admiralty Way/Marina del Rey/California 90292-6695

This research is supported by the Defense Advanced Research Projects Agency under Contract No. MDA903 81 C 0335. Views and conclusions contained in this report are the author's and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. Government, or any person or agency connected with them.

1. Introduction

The evolution of large electronic mail systems testifies to the increasing importance of electronic mail as a means of communication and coordination throughout the scientific research community. These systems include the DARPA Internet mail system, the GTE Telemail system, the MCI Mail system, and the IEEE Compmail system. Until now these systems have operated autonomously, and no convenient mechanism has existed to allow users of one system to send electronic mail to users on another system. INTERMAIL is an experimental mail forwarding system which allows users to send electronic mail across such mail system boundaries. The use of INTERMAIL is transparent, in that users on each system are able to use their usual mail programs to prepare, to send, and to receive messages. No modifications to any of the mail programs on any of the systems are required.

This paper describes the implementation of the INTERMAIL system and discusses some of the problems associated with the interconnection of mail systems. The following sections provide

- background information on the various mail systems using INTERMAIL
- an overview of the current INTERMAILimplementation
- alternative approaches to addressing in a mail forwarding system
- a discussion of the INTERMAIL user interface
- a review of the open issues in the area of mail forwarding.

2. Background

The DARPA Internet mail system is made up of a number of hosts, connected by mail delivery programs that utilize the Simple Mail Transfer Protocol (SMTP). Each of these hosts contains various user mailboxes. For historical reasons, a DARPA Internet mail system mailbox is often referred to as an "ARPANET mailbox". It should be noted that these mailboxes may be located on any host which is a part of the Internet. An ARPANET mailing address consists of two parts, a mailbox name and a host name, separated by the "@" character. For example, a user named "John Smith" at the Internet host "HOST-XYZ" might have the ARPANET mailing address "Smith@HOST-XYZ". Consistent with this convention, the ARPANET mailing address for the INTERMAIL system is "INTERMAIL@USC-ISIF".

A user mailbox is usually a file that contains the messages that a user has recieved. On a TOPS-20 system such as USC-ISIF, a person's mailbox is a file named "MAIL.TXT" in his login directory. These messages are appended to the mailbox by the mail distribution component of the mail system. The user interface component of a mail system, or mail program, allows the user to compose and to send messages. It also enables the user to perform mailbox management functions such as reading new messages and deleting old messages. When a message is being composed, the user interface program prompts the user for the header information and the text of the message. Many mail programs contain a built-in text editor. Once the user has indicated that the message is complete by entering a "SEND" command, the mail program turns the message over to the mail distribution component of the system.

Figure 1 illustrates the path of a message sent by a user on one Internet host to a user on a remote Internet host. The user sends mail using the mail program. The message is picked up by the local

mailer program to be delivered to the mailer program on a remote host. The remote mailer program appends the message to the mailbox of the addressee. The user mail program can be used to read, to file, and to delete messages in a user mailbox.

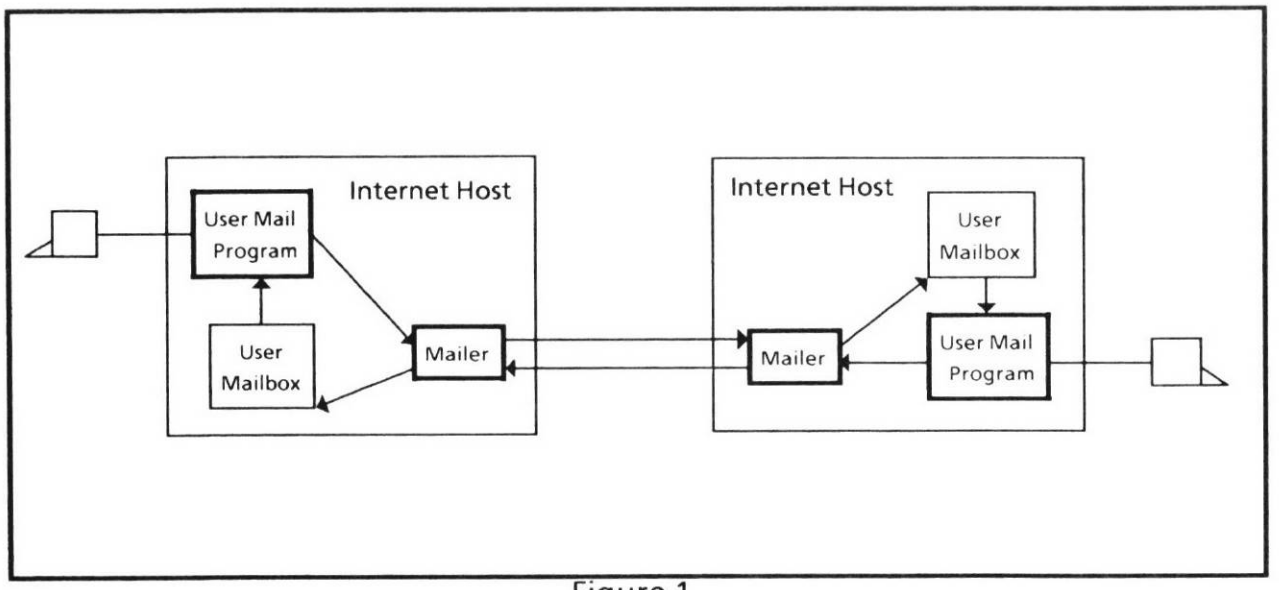


Figure 1

A message consists of a header and text. The header section of a message consists of various fields. Some of these fields, such as the "To" field and the "Cc" field, are used by the mail system to determine the destination(s) of the message. Others, such as the "Date" field, the "Subject" field, and the "From" field are provided for the information of the recipient. In many mail systems the "Date" field and the "From" field are provided automatically by the mail system, as the message is sent.
The following is an example of a typical ARPANET message.

```
Date: 12 Feb 1985 11:17 PST
Subject: SNDMSG Problem
From: Anna-Lena Neches <ALNECHES@USC-ISID>
To: DESCHON@USC-ISIF.ARPA, Postel@USC-ISIF.ARPA
Cc: ALNeches@USC-ISID.ARPA

Hi Annette,

There was a problem with our HOSTS3.BIN-file which was installed
last night.  I have temporarily rolled back the latest version of
the host tables and reinstalled the version from yesterday......

Regards,

Anna-Lena
```

The commercial systems (MCI Mail, Telemail, and IEEE CompMail) all have a similar model of operation. Each system has a user mail program that is used to send mail. There is a mailbox in which mail is collected, whether the user is logged on or not. When the user logs on, he is informed of any new unread messages. At this point, he may use the user mail program to read messages, to delete messages, to file messages for future reference, or to send a reply. The most significant difference between the commercial systems and the Internet mail system is that the commercial systems generally assume that the user mailboxes are all on the same machine.

3. The INTERMAIL System

The INTERMAIL system currently runs on USC-ISIF, a DEC TOPS-20 system that is a host on the DARPA Internet. Figure 2 illustrates the relationship between INTERMAIL and the mail systems involved. INTERMAIL communicates with each mail system using the same mail programs that a human user would use. Since it uses standard user-interface programs on each system, it does not have to be involved in the inner workings of any mail system.

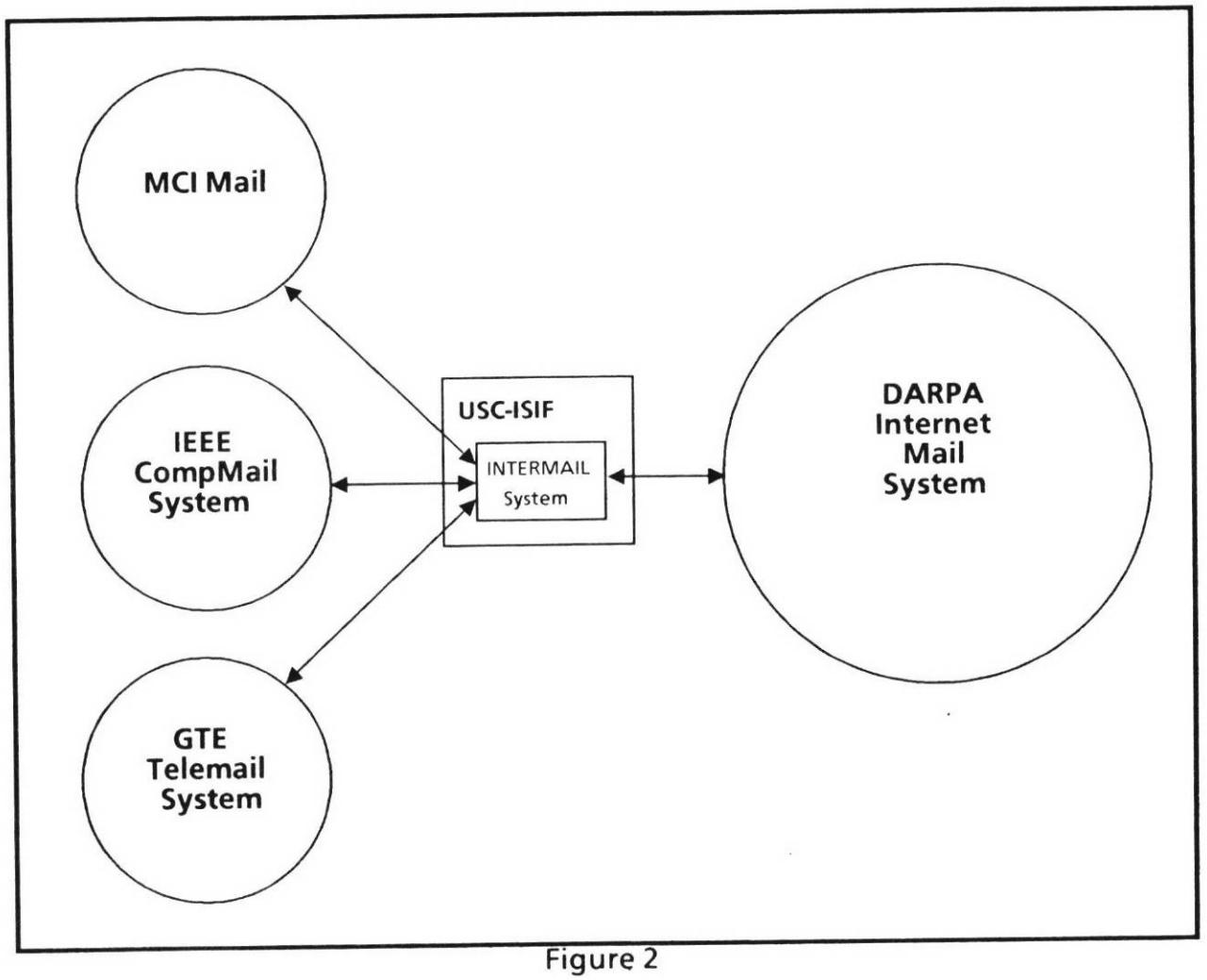


Figure 2

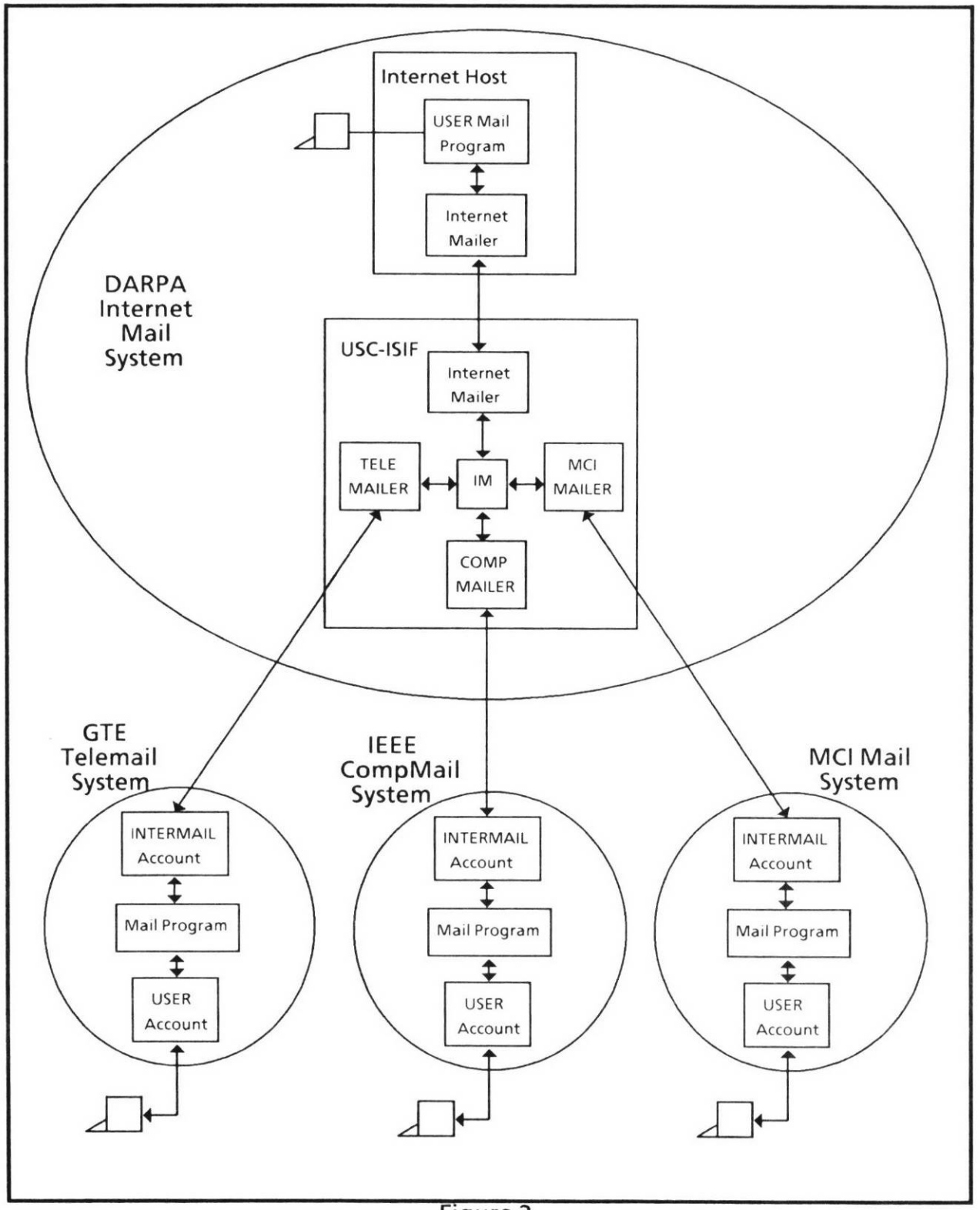

Internet Host
USER Mail Program
Internet Mailer
DARPA Internet Mail System
USC-ISIF
Internet Mailer
TELE MAILER
IM
MCI MAILER
COMP MAILER
GTE Telemail System
INTERMAIL Account
Mail Program
USER Account
IEEE CompMail System
INTERMAIL Account
Mail Program
USER Account
MCI Mail System
INTERMAIL Account
Mail Program
USER Account

Figure 3

Figure 3 illustrates the path that a message takes between a user on a host in the Internet and a user on a commercial system. INTERMAIL consists of several program modules. TELEMAILER, COMPMAILER, MCIMAILER, and the Internet mailer are used to transfer messages between USC-ISIF and other systems. The IM program examines the forwarding information contained each message to determine the destination system. The scheduling of these modules is coordinated by a batch job that runs several times each day. INTERMAIL has a login account and a mailbox on each of the commercial mail systems. A user on a commercial system sends mail destined for a user on a host on the Internet to the "Intermail" mailbox on his local system. INTERMAIL periodically picks up this mail, determines the destination ARPANET address and turns it over to the local Internet mailer for delivery to the appropriate Internet host. In the other direction, a user on an Internet host sends mail destined for a user on a commercial mail system to "INTERMAIL@USC-ISIF". INTERMAIL periodically reads messages from its mailbox on USC-ISIF, determines the destination mail system, and sends the message using TELEMAILER, COMPMAILER, or MCIMAILER.

The programs that make up INTERMAIL include one program that parses the forwarding information on each message, and several mail interface programs that send and receive mail for a particular mail system. The IM program examines the forwarding information contained each message that enters the system, and reformats it for processing by the appropriate mail interface program. If any message contains bad forwarding information or an address that is not accepted by the destination mail system, the program that encounters the problem will copy the message into a "BAD-MSG" file. These messages which contain errors are processed later by a human postmaster, who hopefully can determine what went wrong, and contact the originator of the message. The five mail interface programs are MM, SEND-MSGS, TELEMAILER, MCIMAILER, and COMPMAILER. MM is a TOPS-20 mail program that is used to interface to the "INTERMAIL" ARPANET mailbox. The MM program is used to move mail out of the INTERMAIL mailbox into separate files. The SEND-MSGS program picks up the mail which is destined for an ARPANET mailbox and sends it using the MM program. The TELEMAILER program exchanges messages with the TeleNet mail system (Telemail) using a permanent leased line. The COMPMAILER program exchanges messages with the IEEE CompMail system using a permanent leased line. The MCIMAILER program exchanges messages with MCI Mail using a dial-out modem. We chose to implement TELEMAILER, COMPMAILER, and MCIMAILER in the SAIL programming language because of the built-in text handling functions.

Figure 4 illustrates the communication between the INTERMAIL program modules, through the use of files that are written and read from a common directory. "INTERMAIL Mailbox" is an ordinary MAIL.TXT file in the <INTERMAIL> directory on USC-ISIF. The other files shown contain messages (one per file) at various stages in the forwarding process. A batch job that logs into the <INTERMAIL> directory on USC-ISIF is used to run these programs in the following order:

1. MM reads messages from INTERMAIL's ARPANET mailbox into "IN1" files.

2. IM converts "IN1" files to "(--UNSENT-TELEMAIL--)", "(--UNSENT-MCIMAIL--)", "(--UNSENT-COMPMAIL--)", or "(--UNSENT-ARPAMAIL--)" files.

3. TELEMAILER sends messages from "(--UNSENT-TELEMAIL--)" files to Telemail and reads messages from the Telemail "INTERMAIL" account into "IN1" files.

4. MCIMAILER sends messages from "(--UNSENT-MCIMAIL--)" files to MCI Mail and reads messages from the MCI Mail "INTERMAIL" account into "IN1" files.

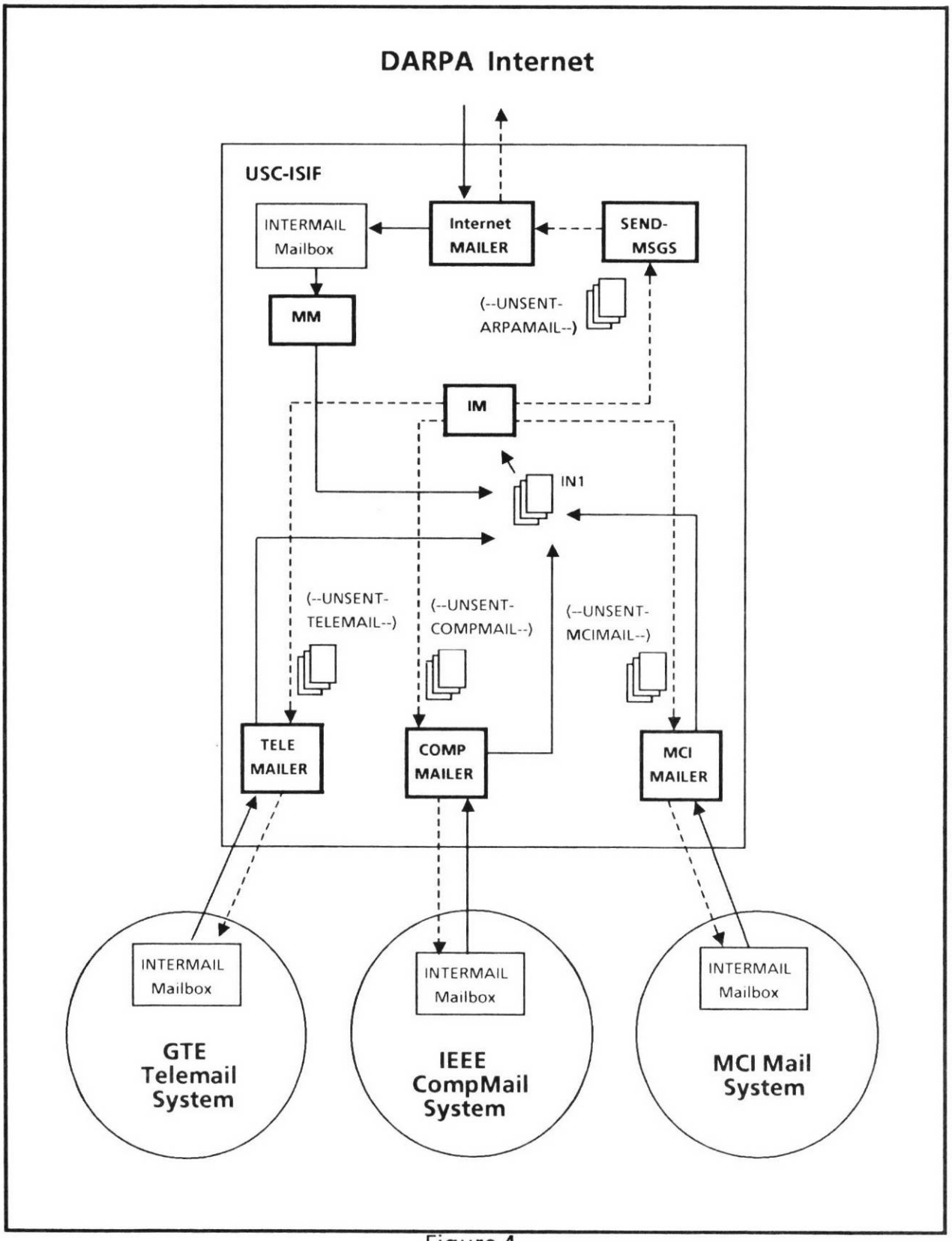

DARPA Internet
USC-ISIF
INTERMAIL Mailbox
Internet MAILER
SEND-MSGS
MM
(--UNSENT-ARPAMAIL--)
IM
IN1
(--UNSENT-TELEMAIL--)
(--UNSENT-COMPMAIL--)
(--UNSENT-MCIMAIL--)
TELE MAILER
COMP MAILER
MCI MAILER
INTERMAIL Mailbox
GTE Telemail System
INTERMAIL Mailbox
IEEE CompMail System
INTERMAIL Mailbox
MCI Mail System

Figure 4

5. COMPMAILER sends messages from "(--UNSENT-COMPMAIL--)" files to CompMail and reads messages from the CompMail "INTERMAIL" account into "IN1" files.

6. IM converts "IN1" files to "(--UNSENT-TELEMAIL--)", "(--UNSENT-MCIMAIL--)", "(--UNSENT-COMPMAIL--)", or "(--UNSENT-ARPAMAIL--)" files.

7. SEND-MSGS sends messages from "(--UNSENT-ARPAMAIL--)" files, destined for ARPANET mailboxes, using the MM program.

The commercial mail systems that are connected by the INTERMAIL system were all designed to be used by human users, as opposed to other computer programs. Therefore, these "user friendly" interface programs tend to use English sentences, which may or may not be phrased exactly the same way from day to day. In addition, line noise is sometimes a problem when the connection to the remote system is via a dial-out modem. The most challenging aspect of these implementations has proved to be recovering from a situation in which a prompt or an error message has been misunderstood.

4. Addressing

The addressing information used by Intermail is included at the beginning of the text of the message. This approach was chosen largely in order to facilitate the use of the existing commercial systems. Each of the mail systems connected by INTERMAIL has its own addressing conventions:

- An ARPANET mailing address consists of a mailbox name and a host name, separated by the "@" character, for example, "Smith@USC-ISIF". Mailbox names are set up to be unique within each Internet host.

- An MCI Mail address consists of a unique seven digit user id, for example, "123-4567". The corresponding user name, for example "JSmith", may be used instead, but user names are not guaranteed to be unique, so the sender may be asked to choose the correct user id from a from a list of possible matches.

- A CompMail address consists of a host number and a user id separated by the ":" character, for example, "64:CMP0123". In addition, a user name, for example "J.Smith", may be used so long as it is locally defined for the sender's login directory.

- A Telemail mailing address consists of a user name and one or more organization names separated by the "/" character, for example, "JSMITH/ACCTG/XYZCORP". An abreviated form, for example "JSMITH", may be used so long as it is locally defined.

The INTERMAIL system has an account on each of these mail systems. Its ARPANET mailing address is "Intermail@USC-ISIF". On MCI Mail it is "107-8239" or "Intermail". On CompMail it is "64:CMP0817" or "Intermail". On Telemail it is "INTERMAIL/USCISI". Therefore on each of these systems, the messages that are to be forwarded to a user on an Internet host are addressed to "Intermail" in the header section of the message.

The INTERMAIL system requires forwarding information consisting of the name of the destination mail system and the name of the addressee, as he is known in the destination system. In setting up forwarding in the CompMail-to-ARPANET direction, we encountered a minor problem with the

character set interpretation. The "@" character, which is used in every ARPANET mailing address, is also used as the default line-delete character on the CompMail system and therefore must be reset before any messages containing an ARPANET mailing address can be composed.

Another approach to addressing is to embed the address of the recipient on the remote mail system in the "mailbox name" part of the ARPANET mailing address. Such an address might be "JSmith%MCIMail@HOST" or "J.Smith%CompMail@HOST". We call this the relay approach. Certain modifications to the mail programs on the host running INTERMAIL, would make this approach possible in the ARPANET to commercial system direction. It would, however, be impossible in the other direction, since there is no way to modify the addressing scheme used in a commercial system.

One advantage to the relay approach is that it is possible to include users from different mail systems on the same ARPANET mailing list. Another advantage is that the "reply" command in the user interface program can be used to send a message back to the originator of the message being replied to, even when the message came from a different mail system. The disadvantage is that if this approach were adopted, the addressing conventions would no longer be the same in both directions. In addition, it would be impractical to make modifications to all of the TOPS-20 mail programs.

5. The User Interface to INTERMAIL

Our goal was to design a user interface that is flexible enough to accept the addressing information in a form familiar to the sender. In order to insure that the header information from the source system can be mapped into the header information of the destination system, INTERMAIL interprets only the header fields that are common to all four mail systems. These fields are the "To" field, the "Cc" field, the "Subject" field, and the "From" field. The header fields that can be delivered to the end recipient unchanged, are said to have end-to-end significance. In this system the fields that are in this category are the "Subject" field and the "From" field. The syntax for a list of addressees for the "To" field and the "Cc" field vary from system to system. In the Internet mail system, the addressees are entered in a list, separated by commas. Continuation lines begin with at least one space.

```
To: Smith@HOSTA, Jones@HOSTB, Lewis@HOSTC, Scott@HOSTD,
    Williams@HOST1, York@HOST2
Cc: DeSchon@USC-ISIF, Postel@USC-ISIF
```

In MCI Mail, the addressees are entered on separate lines. If a postal address or a telex address is used, the address requires more than one line.

```
To: AJones
To: John Smith (ADDRESS)
    1234 Maple Street
    Los Angeles, Ca. 90025
To: 123-4567
CC: JWashington
```

In the CompMail system, addresses are entered in a list separated by either commas or spaces. There is no separate "Cc" field. In addition, the "&" character is used to signify that a line is being continued.

```
To: J.Smith A.Jones F.Lewis CC J.Washington &
    L.Ford S.Chan M.Williams
```

In the Telemail system addresses are entered in a list separated by commas.

```
To: JSMITH/NASA, AJONES/XYZCORP, FLEWIS/USCISI
Cc: JWASHINGTON/XYZCORP
```

It is therefore necessary that INTERMAIL accept addresses in the format that is used on the destination mail system. Some allowances for the practices employed on the source mail system are made when there is no confict. For example, when a message destined for CompMail contains an explicit "Cc" field, or uses the Internet conventions for a continuation line, INTERMAIL reformats the addressees into a form that is acceptable to the CompMail user interface program.

Two formats can be used to specify the addressing information in the forwarding section: "Simple Forwarding" and "Source Route Forwarding". INTERMAIL originally supported only Source Route Forwarding. It is more general than Simple Forwarding, in that it provides the mechanism for multi-hop source routing. Simple Forwarding was later designed for easier use and it will handle most cases.

5.1. Simple Forwarding

In Simple Forwarding, the forwarding section contains the name of the destination mail system and the address(es) of the recipient(s) of the message, in the form

```
Forward: <destination-mail-system>
To: <recipient's-address>
Cc: <someone-else's-mailing-address>
<blank line>
```

The IM program module processes the "Forward" field to determine which mailer program (SEND-MSGS, MCIMAILER, COMPMAILER, or TELEMAILER) will send the message. It then uses the "Subject" field from the header, and the addressing information from the forwarding section to make up a new header for the unsent mail file. At this point it deletes the forwarding section from the text, replacing it with the "From" field of the original header, and copies the text of the message into the unsent mail file. The header section that the recipient sees when the message is ultimately delivered is the header section that is created when the message is sent by the appropriate mailer program. For example, the following ARPANET message

```
Date: 18-May-84 11:52:21 PST
To: Intermail@USC-ISIF
From: Annette DeSchon <DeSchon@USC-ISIF>
Subject: Sample Message

Forward: MCIMAIL
To:123-4567

This the text of the message.
```

might be delivered to the MCI Mail recipient as

```
Date:      Fri May 18, 1984 11:58 am  PST
From:      Intermail / MCI ID: 107-8239

TO:      * John Smith / MCI ID: 123-4567
Subject:   Sample Message

From: Annette DeSchon <DeSchon@USC-ISIF>

This the text of the message.
```

5.2. Source Route Forwarding

Source Route Forwarding was developed to handle situations in which a message will travel multiple hops before reaching its destination. As in Simple Forwarding, the forwarding information is included at the beginning of the text of the message. Instead of a single destination mail system and address list, the forwarding section contains a series of Forward Source Route (FSR) lines, which specify the path that the message will take as it is being forwarded. The first FSR is the Internet mailing address of the mail forwarding program. The second FSR is used by the IM program to create the address section of the message for the next hop that the message will take. The forwarding section must contain at least one Return Source Route (RSR) line which ultimately may be used by the end recipient to create the forwarding section of a reply. Each FSR or RSR consists of a network or mail system name and the name of a user as he is known in the destination network. The network of the first FSR must match the network of the first RSR. This is a simple example of the source routing section in a message which is to be forwarded to MCI Mail.

```
FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]<recipient's-MCI-address>
RSR: [ARPANET]<sender's-Internet-mailing-address>
```

A multi-line format FSR or RSR may be used when a header field does not fit onto a single line. This format is used to enter a postal address (for MCI Mail), or when the list of recipients is too long for one line. The multi-line format also allows the creator of the message to specify optional fields such as the "Cc" field, and the "Subject" field. In a multi-line format FSR or RSR, the first line contains only the label and the network or mail system name. The lines following, up to the next FSR or RSR, will be copied into the appropriate unsent mail file (e.g., "(--UNSENT-MCIMAIL--)"). Therefore, the syntax of these lines will be similar to the syntax for the corresponding fields in the destination mail system. This is an example of a forwarding section containing a multi-line format FSR.

```
FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]
To: <recipient's-mailing-address>
Cc: <someone-else's-mailing-address>
RSR: [ARPANET]<sender's-Internet-mailing-address>
```

As a message is forwarded, the forwarding section is modified by the IM program to reflect each hop that the message has taken. Therefore, the forwarding section of the message as it is received by a user on the destination system is not the same as it was when it was originally sent. In addition, the header section (or "envelope") that the recipient sees is the header section that is created when the message is sent by the mail interface program, rather than the original header. For example, the following ARPANET message

```
Date: 18-May-84 11:52:21 PST
To: Intermail@USC-ISIF
From: DeSchon@USC-ISIF
Subject: Sample Message

FSR: [ARPANET]Intermail@USC-ISIF
FSR: [MCIMAIL]123-4567
RSR: [ARPANET]DeSchon@USC-ISIF

This the text of the message.
```

might be delivered to the MCI Mail recipient as

```
Date:     Fri May 18, 1984 11:58 am  PST
From:     Intermail / MCI ID: 107-8239

TO:     * John Smith / MCI ID: 123-4567
Subject:  Sample Message

FSR: [MCIMAIL]123-4567
RSR: [MCIMAIL]Intermail
RSR: [ARPANET]DeSchon@USC-ISIF

This the text of the message.
```

Figure 5 illustrates a hypothetical situation in which Source Route Forwarding is used to accomplish multi-hop forwarding. So far, in practice all of our forwarding is accomplished in a single hop. This example is included to demonstate how Source Route Forwarding would work in a situation where Simple Forwarding could not be used. In this example, each mail system uses an addressing convention different from the addressing convention used by the other mail systems pictured. (In actuality these differences can take the form of disagreement about what a host name looks like, what a user name looks like, or the syntax used to express "user at host".)

In the example pictured in Figure 5, host names in Mail System A must begin with a letter. Host names in Mail System B must begin with the characters "!!", and in Mail System C, host names must begin with the characters "$$". Since hosts XYZ and ABC are each connected to two mail systems which use different addressing conventions, the source route from Person1 at host XXX to Person2 at host ZZZ will contain different address names than the source route which specifies the opposite

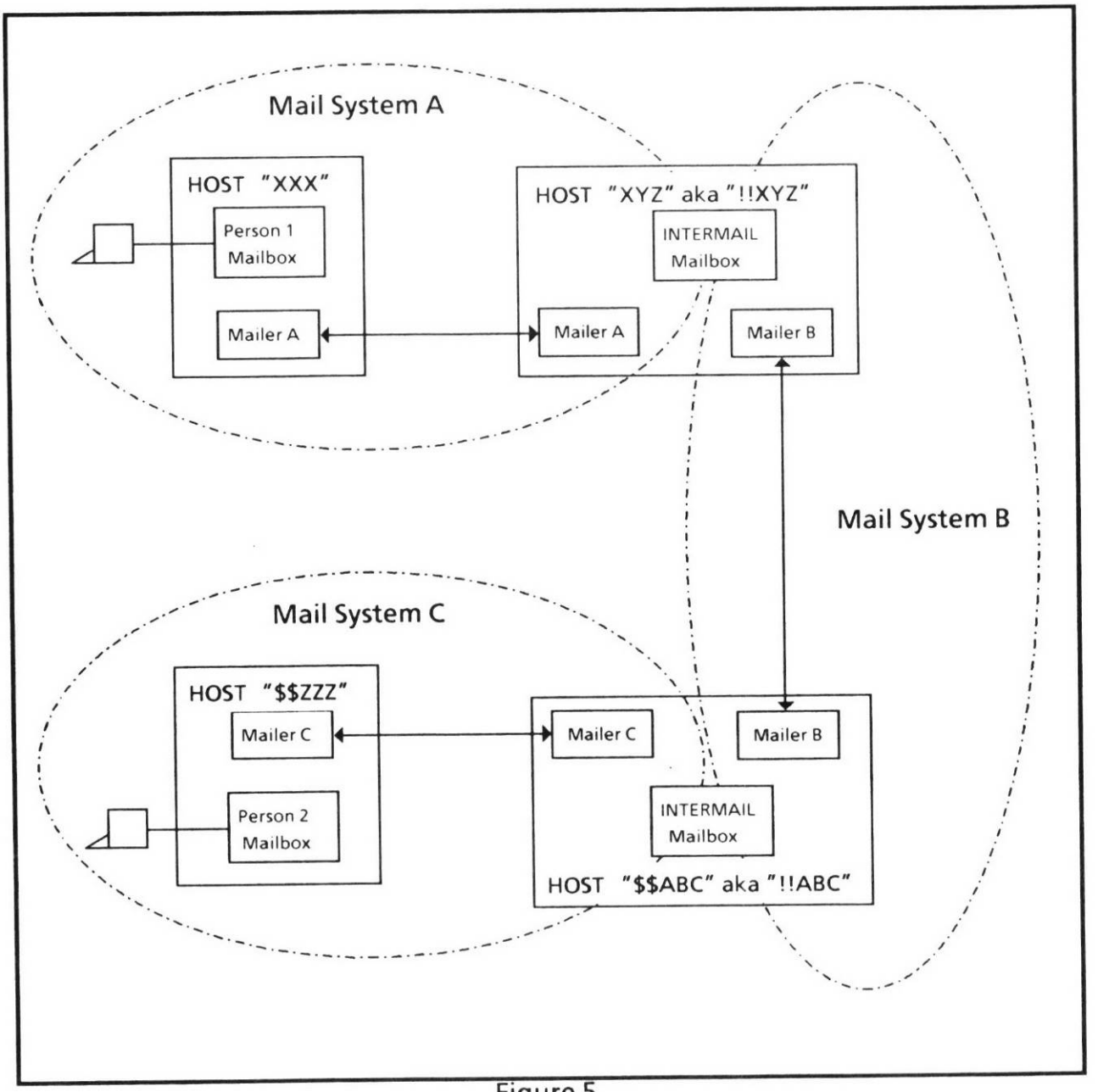


Figure 5

direction. When Person1 at host XXX sends the message to the forwarder at host XYZ, the forwarding section of the message consists of

```
FSR: [A]Intermail@XYZ
FSR: [B]Intermail@!!ABC
FSR: [C]Person2@$$ZZZ
RSR: [A]Person1@XXX
```

The forwarder at host XYZ then deletes the first entry from the Forward Source Route list, and inserts its return address at the beginning of the Return Source Route list. When the message is sent from

host XYZ to the forwarder at host ABC, the forwarding section consists of

```
FSR: [B]Intermail@!!ABC
FSR: [C]Person2@$$ZZZ
RSR: [B]Intermail@!!XYZ
RSR: [A]Person1@XXX
```

Next, the forwarder at host ABC performs the same operation on the source route, and sends the message to Person2 at host ZZZ. At this point the forwarding section consists of

```
FSR: [C]Person2@$$ZZZ
RSR: [C]Intermail@$$ABC
RSR: [B]Intermail@!!XYZ
RSR: [A]Person1@XXX
```

In order to reply to a message that has been forwarded using this means, the following procedure may be used to construct the source route. Starting with the message as it is received, exchange the FSRs for RSRs, and the RSRs for FSRs. Then place the FSRs in front of the RSRs, at the beginning of the source routing section. This yields the following source route

```
FSR: [C]Intermail@$$ABC
FSR: [B]Intermail@!!XYZ
FSR: [A]Person1@XXX
RSR: [C]Person2@$$ZZZ
```

At this point Person2 would send the message to "Intermail@$$ABC" and the same steps would take place in the other direction.

6. Areas for Future Study

Thus far, the INTERMAIL system has been developed and operated as an experiment. Should we decide to establish an actual service, the issues of billing and access control will have to be addressed. In addition, more can be done in the area of automating the postmaster's job. We would like to improve the program-generated error messages and possibly develop the capability to return messages containing addressing errors automatically.

The most commonly encountered problem is that users completely omit the forwarding section of the message. This error is especially common when the user is replying to a message. This is probably because the "reply" command completely takes care of all of the addressing requirements when the reply is to a sender within the same mail system. To improve the situation on the Internet side we would like to establish a user interface program that would prompt the user for the forwarding information, and insert it into the text in the correct format. Using this approach it will also be possible to automatically construct replies to messages that have been forwarded to the Internet mail system.

7. Acknowlegements

The INTERMAIL forwarding system is based on work done by Danny Cohen, Jon Postel, Joel Goldberg, and Lee Richardson. It was originally developed to communicate with users of the MOSIS VLSI system via the Telemail system.

8. References

[1] D. Cohen, "A Suggestion for Internet Message Forwarding for MOSIS", IEN 180, USC Information Sciences Institute, March, 1981.

[2] D. Cohen and J. Postel, "Gateways, Bridges, and Tunnels in Computer Mail", Technical Report ISI/RS-83-117, USC Information Sciences Institute, January, 1984. Also appears in "Local Networks: Distributed Office and Factory Systems", Procedings of Localnet '83, New York, and "Local Networks: Strategy and Systems", Procedings of Localnet '83, New York.

[3] D. Crocker, "Standard for the Format of ARPA Internet Text Messages", RFC 822, University of Delaware, August, 1982.

[4] A. DeSchon, "MCI Mail / ARPA Mail Forwarding", Technical Report ISI/RR-84-141, USC Information Sciences Institute, August, 1984.

[5] GTE, "Telemail User's Guide", GTE Telenet Communications Corporation, 8229 Boone Blvd., Vienna, Virginia 22180, 1984.

[6] IEEE, "COMPMAIL + User Manual", IEEE Computer Society, P.O. Box 3489, Silver Spring, MD 20901, September, 1983.

[7] MCI, "Basic User's Guide", MCI Mail, 2000 M Street, N.W., Washington D.C. 20036.

[8] MCI, "MCI Mail Service Guide", MCI Mail Customer Service, Box 1001, 1900 M Street, N.W., Washington D.C. 20036.

[9] J. Postel, "Simple Mail Transfer Protocol", RFC 821, USC Information Sciences Institute, August, 1982.

(0) NACC: 2007-85
(1) DOC-TYPE: 4
(3) REPORT-NO: ISI/RR-85-158
ISI RR 85 158
(11) CONTRACT-NO: MDA-903-81-C-0335
MDA 903 81 C 0335
(16) PERSONAL-AUTHOR: DeSchon, A.L.
(17) AUTHOR-AFFIL: University of Southern California, Information Sciences Inst
(22) TITLE: Intermail, an experimental mail forwarding system
Intermail an experimental mail forwarding system
(23) PLACE-PUB: Marina del Rey, CA
(24) PUBLISHER: University of Southern California, Information Sciences Inst.
(26) PUB-DATE-MR: 850900
(27) PUB-DATE: 1985 September
(28) PAGES-TOTAL: 18 p
(67) MEDIA-TYPE: p
(70) PRIM-SEC-REF: 2
(74) KEYWORDS: ARPANET
Advanced Research Projects Agency Network
Compmail
Computer mail
Intermail
IM
Mail forwarding
MCI mail
GTE telemail
MM
Electronic mail
Heterogeneous mail systems
Intermail address
Intermail header
Intermail user interface
(75) ABSTRACT: Intermail (IM) is an experimental mail forwarding program that enables users on one mail system to send mail to people on another, previously autonomous, mail system. The mail systems it interconnects are the DARPA (Defense Advanced Research Projects Agency) Internet mail system (also called MM), the GTE Telemail system, the MCI Mail system, and the IEEE Compmail system. Messages from each system which are bound for another system are sent to "Intermail" at a host at ISI (Information Sciences Institute). There, the Intermail program analyzes the address and forwards the message to the proper recipient on the proper network. The details of addressing, the user interface, and other aspects of the Intermail system are given. Areas for future study are discussed.
(76) ENTRY-DATE: 860220
(77) UPDATE: 871209-145545 APRIL
(82) DIST: Approved for public release. Distribution unlimited.
(83) CPLACE: Marina del Rey, CA
(85) LOCATION: Main
(86) BIBSET: General Net
(88) CORP-ORG: University of Southern California, Information Sciences Inst.

This paper describes an architectural approach that provides information interchange across a broad spectrum of user applications and office automation offerings. Some of the architectures described herein are currently implemented in existing IBM products. These and other architectures will provide the basis for document interchange capability between products such as the IBM 5520 Administrative System, the IBM System/370 Distributed Office Support System (DISOSS), and the IBM Displaywriter System. Specifically described is a document distribution architecture and its associated data streams. Transforms can be utilized to interchange between these data streams and others.

A general overview of the architectures as opposed to a detailed technical description is provided. The architectures described are protocols for interchange between application processes; they do not address the specific user interface. The document distribution architectures utilize SNA for data transmission and communications control facilities.

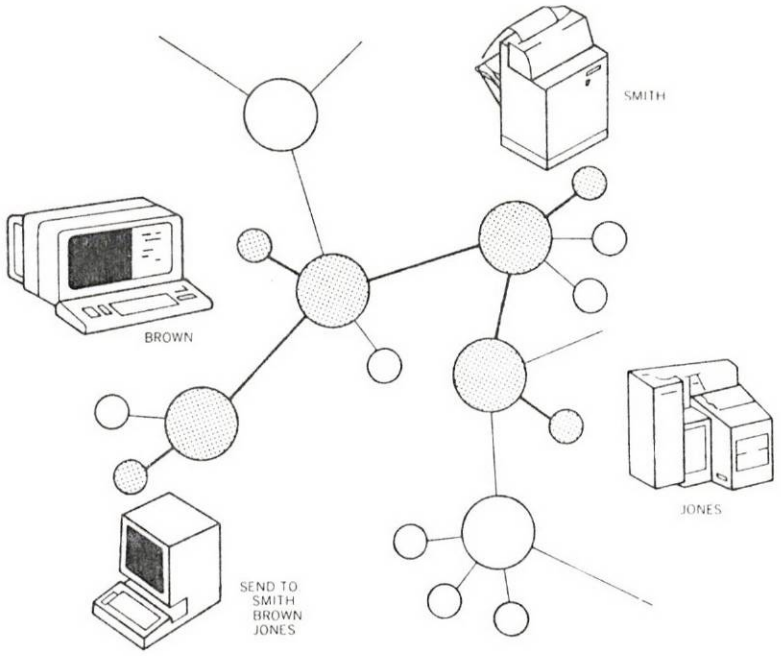
Electronic information interchange in an office environment

by M. R. DeSousa

The desire to interchange office information electronically has been with us since Samuel Morse invented the telegraph in the 1840s. Today, "electronic document distribution" is a business buzzword; professional journals and trade magazines abound with references to the automated office, electronic document distribution, and communication networks. The office-information-interchange system is fast becoming a reality.

The office-information-interchange system is envisioned to work in a network such as the one shown in Figure 1. Such a network is a complex interconnection of systems of various capabilities and a large number of different terminals, or work stations, performing an assortment of applications. A major problem that such a network presents is the variety of interfaces and data forms that must be accommodated to functionally interconnect such devices into an operational information-interchange system.

Figure 1 Office information interchange network



system requirements

Not only must the devices be functionally interconnected, but the office-information-interchange system must provide the basic capability to:

- Enter and edit information
- Distribute information
- Print (or display) information

The "enter and edit" capability allows an originator to create information that must be portable between devices of differing functional characteristics while maintaining the capability to perform additional editing and revision.

The "distribute" capability requires a set of processes that communicate directions and interchange control information and carry the information from an originator, or sender, to a recipient.

The "print" capability, in this context, implies a sender/recipient relationship where the sender wants to distribute information in its final form to the recipient. It is analogous to a traditional mail environment. It is assumed that the recipient has no need to modify or edit the information. To ensure that the information conveys its intended meaning, the sender optionally should have assurance that the information will be printed exactly as specified by the sender (Figure 2).

Figure 2 Print fidelity (Product A creates and prints a document. Because the document is in columnar format, a substitution to the font or space requirement could totally change the meaning of the information. Therefore, when Product A interchanges the document, Products B and C must print the document exactly as Product A did.)

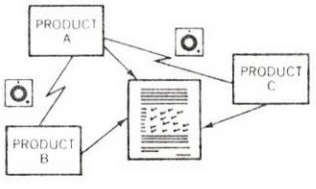
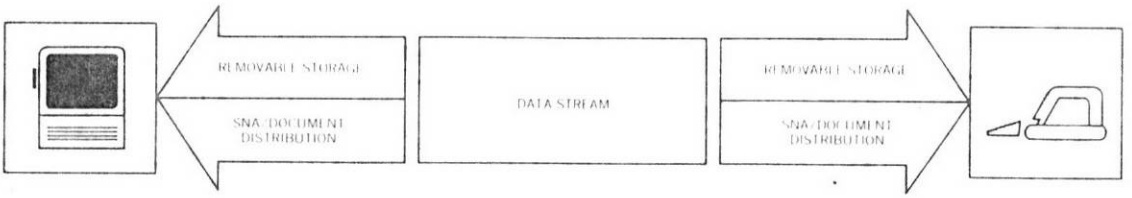


Figure 3 Information interchange system architectures



It is not necessary that each device within the network provide all three basic capabilities. For example, a device that is capable of distributing information need not be capable of creating nor printing it.

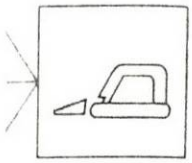
Irrespective of the configuration, the system must be easy to use if it is to be effective. The complexities of interfaces and data forms must be transparent to the users. To expect a sender of information to know the device and data-form requirements is unrealistic. The sender should be able to request that the information be distributed and should not have to be concerned about the devices used by the recipients.

The approach

A set of architectures has been defined that specifies the control of information (referred to as the data stream), the document distribution application, the communication transmissions, and any removable storage media being used to manually interchange information (Figure 3). The data stream architectures specify the form of the information by describing the syntax and meaning of allowable elements in the data stream.

The document distribution, or interchange, architecture permits information to be carried from an originator to a recipient without requiring that both be interactively communicating during the distribution process. Further, it allows an originator to send information (a document) to multiple recipients with a single distribution request. And, finally, the distribution architecture provides for services such as security, safe storage during the distribution process, and confirmation of delivery.

The architecture required for transmission control already exists, and it is IBM's Systems Network Architecture (SNA).[1-5] For information interchange via removable storage media, there is a requirement for an interchange diskette architecture. Neither SNA nor the diskette architecture requirement is described in this paper.



A way of putting these architectures into the proper perspective is to liken SNA to a postman, the distribution architecture to an envelope, and the data stream to a message or letter within the envelope.

Because of the unique functional requirements needed to perform both enter/edit and print in a single data stream, it was clear that both capabilities could not be satisfied through a single architecture. The information-creation process requires functional richness which would not easily be transformed into a presentation data stream at a printer or display. The print process requires simplicity if it is to be compatible with a wide range of output devices.

An enter and edit data stream should also be portable between differing devices of unlike functional levels. For example, if information is entered on device A and sent to device B for additional entry and editing, device B should perform identically those functions that both devices support and must recognize those functions it does not support and provide a responsible, appropriate response. With a single data stream, any reasonable approach that would permit interchange on devices of unlike functional levels would preclude guaranteed print fidelity.

A print data stream can be much simpler than an edit and enter data stream because complex editing and formatting functions are not required. This is explained in more detail later. But, with the potential combination of text and other forms of data, the print data stream becomes more complex.

specific DCAs

In order to separate the complexities deriving from differing data stream requirements, it was concluded that architectures for both revisable-form and final-form data streams were required. These architectures are generically referred to as Document Content Architectures or DCAs. The specific DCAs described are the Revisable-Form DCA, the Final-Form Text DCA, and the Final-Form Mixed Data DCA.

The revisable-form data stream is text and has not yet been transformed into its final form. The text, although sufficiently formatted for presentation, can still "flow" as the result of additions and deletions. The revisable-form data stream is used when editing text or when distributing text for revision.

This paper describes only one revisable-form data stream. This is not to imply that this DCA is the only DCA that can be used in an interchange system. Several revisable-form data streams already exist. Some of those generally used are the IBM Document Composition Facility Program Product, the IBM Generalized Markup Language (GML) Program Product, the IBM Script/370, and the

IBM 3730 Distributed Office Communications System text data stream. These data streams are found in IBM's data processing systems. Because these data streams may coexist in an information interchange system with the DCAs described herein, transforms within the system may be required to ensure information interchange of revisable information for distributed document development. IBM's other text-processing (or word-processing) products such as the Displaywriter use the Revisable-Form DCA described herein.

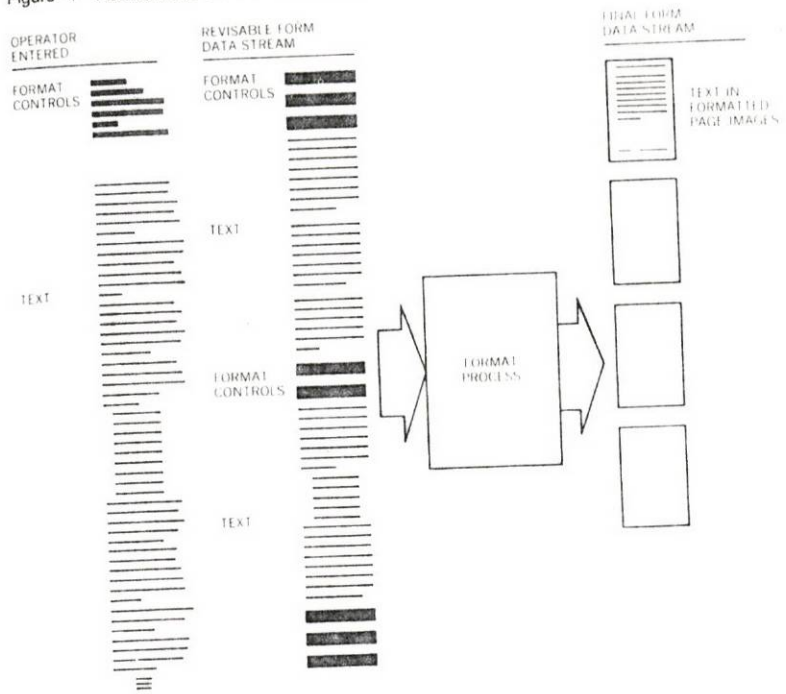
The final-form text data stream is text that has already been formatted and is ready for presentation. This data stream is used where the recipient merely reads the information and files or discards it. It allows the interpretation of graphic characters and of conventional text-processing controls such as carrier return and tab. Such IBM products as the 5520 Administrative System and the Displaywriter use the Final-Form Text DCA described later.

The final-form mixed data type data stream is also formatted and is ready for presentation. Like final-form text, this data stream is intended primarily for the read-only application but differs in that it supports combinations of data types within the same data stream, e.g., text and image.

The final-form DCAs are suitable for use as a device interface format and are independent of any specific device characteristics. All devices that participate in an information-interchange system must support a final-form DCA that guarantees print fidelity when requested.

Figure 4 shows the relationship of the revisable and final-form DCAs. In the revisable form, page-width and page-depth information are carried as general formatting controls at the start of the document and at specific locations within the data stream. Line-ending and page-ending decisions are made to permit viewing during the editing process, but the decisions are subject to change during subsequent processing. The revisable form may also include items such as margin text and pointers to external text for inclusion. The information is not sufficiently formed for final presentation, but it is well-suited for editing because a single change to the margin text, for example, can apply to the complete document. Once the document is transformed into its final form, page and line endings are permanently fixed, and the information is in page image format. Complex formatting controls such as headings and footings, page numbering, and footnotes, which were permitted in the revisable-form data stream, have been resolved and do not appear in the final-form data stream. Again using margin text as the example, when it is placed in its appropriate location on each page as text, a change to the margin text will require a change on every page. It is this transformation of complex for-

Figure 4 Relationship of revisable-form and final-form data streams



matting controls into simple printable text that makes the final-form data streams acceptable to a broad spectrum of output devices. It is this same transformation, however, that makes the final-form data streams unsuitable for editing because the originator's format "intent" is lost.

The discussion of transforms, thus far, has been restricted to the transformation of the revisable-form data stream into the final-form data stream. Because there is more than one data stream in both the revisable-form and final-form applications areas, transforms within each application may also be required. In final form, transformations to and from final-form text and final-form mixed data types will be required. For example, a text-only document distributed as a final-form text data stream will require a transformation when the output is produced on a device whose data stream is final-form mixed data types. In some instances the end user may be involved in the transformation process, but final form transforms should be transparent.

Revisable-Form DCA

The Revisable-Form DCA provides for the interchange of editable documents. Interchange is defined as the proper interpretation of the data stream, but not necessarily complete execution of the function implied or expressed. If a function is performed as speci-

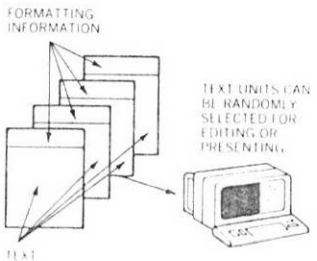
fied or is signaled when not performed, the interpretation is proper. Subsequent editing may then be used to produce a document that may be processed on either the originating or the receiving system.

The means to accomplishing interchange is defined by an architecture that enforces a standard definition of syntax and semantics for reformattable, editable text. The architecture specifies a data stream organization that:

- Guarantees interchange among implementing systems.
- Provides sufficient structure and redundancy in the data stream to support nonsequential access while allowing for identical interpretation whether processed sequentially or nonsequentially.
- Allows format declarations while remaining unformatted, thereby preserving for a user the capability to revise both format declarations and textual content.
- Provides sufficient information so that the document can be printed or displayed in its revisable form (as entered), or, through appropriate processing, in its final-formatted form, or in any form between these two extremes that a system chooses to implement.

The Revisable-Form DCA is specifically designed to support two text-processing environments; *document development* and *distributed document development*. The document development environment is the initial phase of producing a document: the entering of text and format controls into a text-processing system and the editing of individual sections of a document. The distributed document development environment is the distribution of documents to different work stations or text-processing systems because of load balancing or product capability limitations, or to allow revision at a remote location.

Figure 5 Text units in the Revisable-Form DCA



The Revisable-Form DCA supports these applications by defining a common text-processing syntax and semantics and by providing structures for random access of documents on a piecemeal basis, with formatting parameters carried as an integral part of each of these units.

The Revisable-Form DCA data stream is composed of revisable text, embedded text-processing controls, and formatting declarations in the form of structured fields. Each structured field has an *introducer* that contains the length of the declaration, its class, type, and format. This introducer makes all parts of the DCA self-describing.

The content of the revisable document is stored in pieces that are called *text units*. The concept of text units is illustrated in Figure

Figure 6 Re

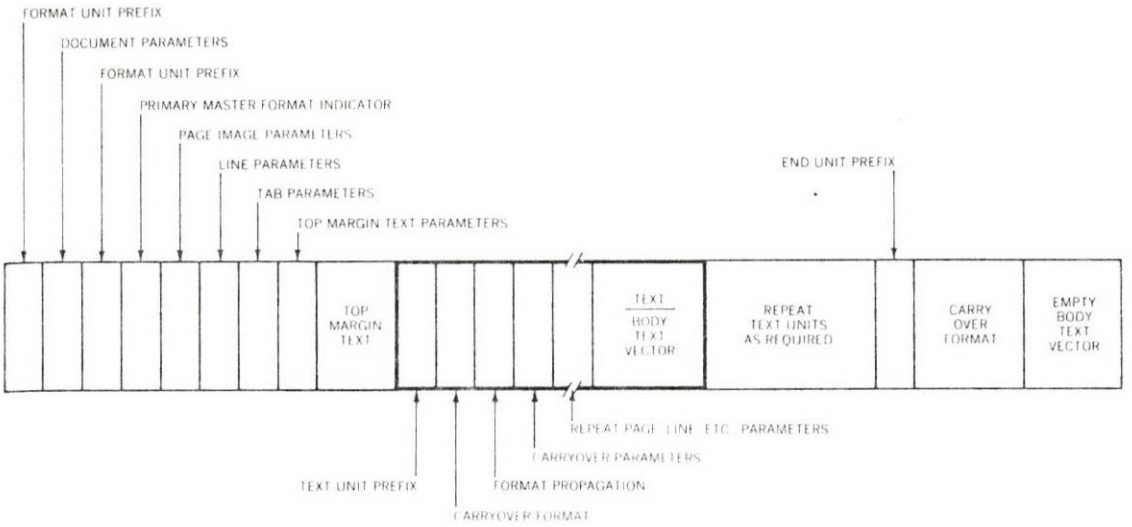
5. Each te
format tha
a text unit
appear if

The revisa
with a te
text unit c
Note that
over form
of the dat

The struc
namic for
be chang
ately. The
inator's "
ample, if
statement
additional

There ma
the revisa
difference
must dete
This requ
the natur
from the

Figure 6 Revisable-form data stream organization



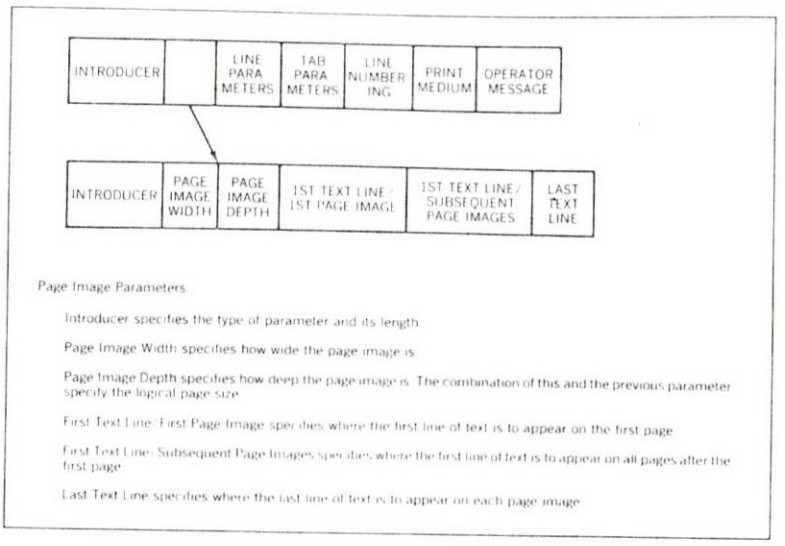
5. Each text unit starts with the formatting parameters required to format that piece of the document. Thus, an originator can select a text unit at random and can display or print that unit as it would appear if the entire document had been processed sequentially.

The revisable-form data stream organization is shown in Figure 6 with a text unit highlighted. The information that precedes the text unit can be thought of as document initialization information. Note that margin text is specified outside of the text unit. A carryover format and empty-body text vector are appended to the end of the data stream for ease of document extension.

The structure of the data stream provides the capability for dynamic formatting while editing at a display; i.e., tab settings can be changed and the effect of the change made visible immediately. The structure also provides the capability to retain the originator's "intent" during entry or editing. Again, by way of example, if the originator adds to or deletes words from a centered statement, the modification is made and centering occurs without additional instruction from the originator.

There may be instances when a device does not fully understand the revisable-form data stream it receives because of functional differences. Because of this possibility, all participating devices must detect differences and handle them as exception conditions. This requirement means that all devices must detect and report the nature and location of the exception condition and recover from the exception condition.

Figure 7 Master format declaration



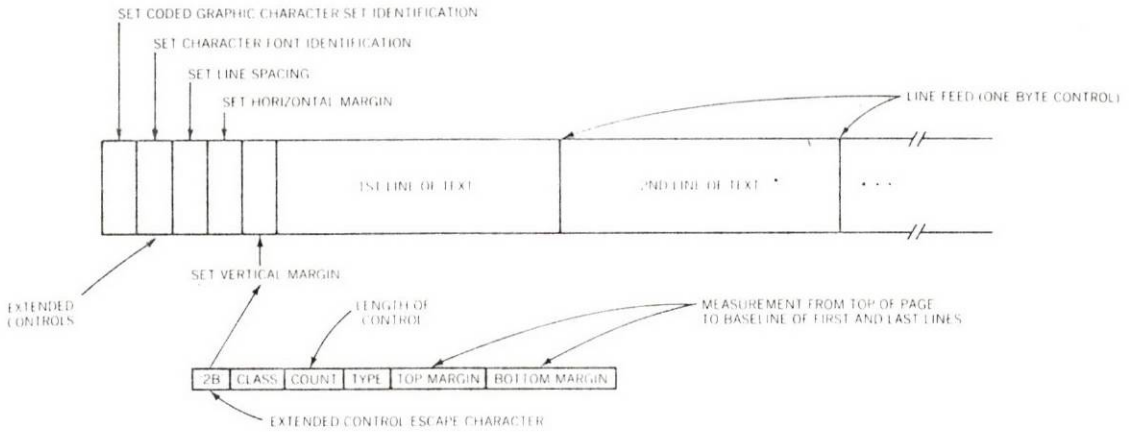
The data stream consists of standard EBCDIC one-byte controls, a set of multiple-byte controls, and a set of self-describing structured fields. It is through use of the structures that the state of the format is maintained and known anywhere within the document and that random access and processing of the revisable text is possible.

One structure—the master format declaration—is described in detail to provide some insight into the architecture. The master format is a statement of the general appearance of page images that will ultimately be put on a display or a piece of paper. Two master formats can be specified in a document: the Primary Master Format and the Alternate Master Format. Figure 7 shows the general content of a master format declaration. The introducer of the declaration specifies whether it is a primary or alternate master format. Within the figure, only the Page Image Parameters field is shown in detail.

Final-Form Text DCA

The Final-Form Text DCA specifies the representation of formatted text information for interchange using communications facilities. This architecture guarantees print fidelity when requested. It is suitable for use as a device interface format and is independent of any specific device characteristics. It provides a simple data stream structure capable of being processed sequentially by synchronous devices.

Figure 8 Final-form text data stream organization



The architecture provides the definition of text and format control function to format and print a document. *Text*, as defined here, means an ordered string of characters (graphic symbols) that are obviously suitable for the specified purpose of representing coherent information. Text is further ordered into units of composition and presentation referred to as *lines*. The lines of text, when assembled into an ordered finite collection, will comprise a presentation unit called a *page*. A single page or a group of pages will comprise a document that is the object or unit of transfer for interchange. The term *print* used here will include displaying a document on a volatile medium such as a video display, reproducing a permanent image on paper or photo-sensitive media, as with impact, ink jet, or photo printers, or recording the document image on magnetic media such as diskettes.

Control functions are designated by specific control codes within the character set used for the text string. The graphic symbols assigned to a text character set are explicitly noncoincident with any of the codes assigned to control functions to prevent obvious ambiguity. The control codes are imbedded within the text at specific positions where a control function is to be activated to produce a desired result with the document in its presented form. Control functions may also activate a state condition for a process algorithm or device action and may be used to instruct an operator about how to operate a device. All the controls that are supported in the Final-Form Text DCA are either EBCDIC formatting controls or extended multibyte controls.

Figure 8 shows the organization of the final-form text data stream. The Set Vertical Margin extended control is shown in detail.

Generally, the EBCDIC one-byte controls provide the basic functions, such as line end, backspace, and indent, and have a one-time, immediate effect. The extended controls are more global and provide for functions such as line spacing, horizontal and vertical margins, and tab setting. A few of the extended controls remain in effect until the line-end control, but most remain in effect throughout the document or until they are reset.

In Figure 2, the concept of print fidelity was introduced. The ability to guarantee exact reproduction is provided through an exception action control. A user can specify what level of deviation he will permit. If no deviation is permitted because of the nature of the information that is to be presented, the user can specify that the information must be presented as specified or the presentation must be terminated. This requirement can be placed on the entire document or at critical points within the document.

Final-Form Mixed Data Type DCA

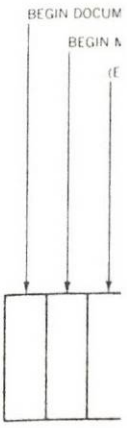
The Final-Form Mixed Data Type DCA consists of a contiguous sequence of structured fields; its structured field organization is similar to that of the revisable-form data stream. Certain of these fields can be grouped into objects that collectively form a document. The objects are bounded by appropriate begin and end structured fields. For example, a document starts with a Begin Document structured field and terminates with an End Document structured field.

The most fundamental object is an elementary data object. This object contains the information to be printed or displayed.

An elementary data object is bounded by a begin-block and an end-block structured field. Within an elementary data object, two additional varieties of structured fields can exist. One contains the actual data, for example, image raster data or text data; the other, which is a collection of structured fields, contains environment control information that describes the autonomous characteristics of the data. The conditions established by the environment control information only apply to the data within the object. The environment is "scoped" by the begin and end structured fields of the object.

An elementary data object is thus a bounded, self-contained unit that is composed of the elementary data to be presented and a complete description of the characteristics of the data. The elementary data object in this form is independent; it is never governed or influenced by data characteristics specified outside of the elementary data object. Only when the data characteristics are not explicitly specified within the elementary data object is

Figure 9 Fin



the data s
outside th

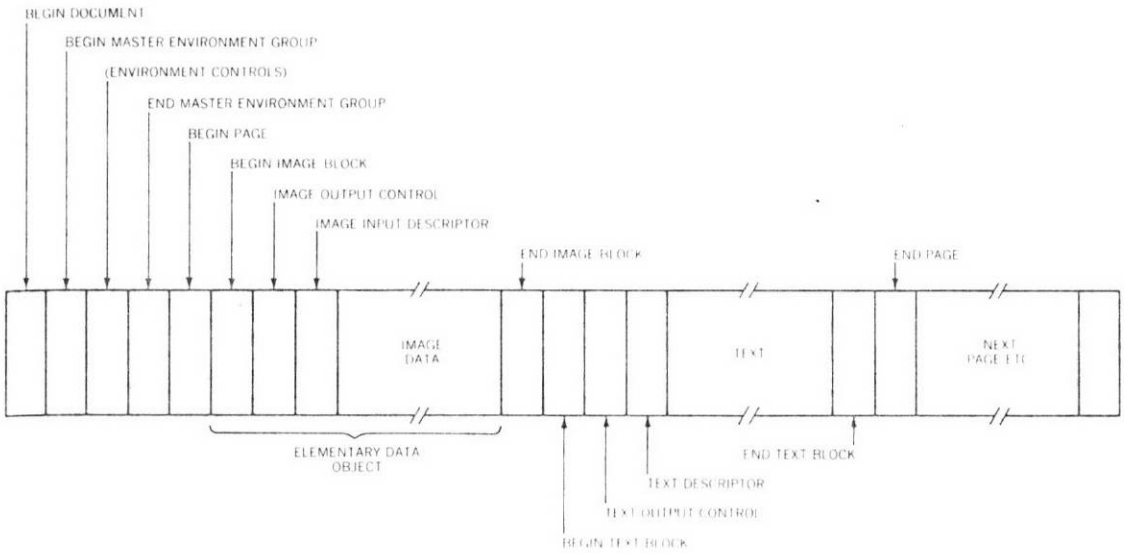
These ele
a page. A
tary data c
ship to the
each othe

The Final-
els: docu
erarchical
levels. Fo
higher lev
single elen

As stated
itly specifi
"scope" is
is not spec
next highe
ing."

To underst
its simples
created by
in size, all

Figure 9 Final-form mixed data type data stream organization



the data subject to the influence of data characteristics specified outside the elementary data object.

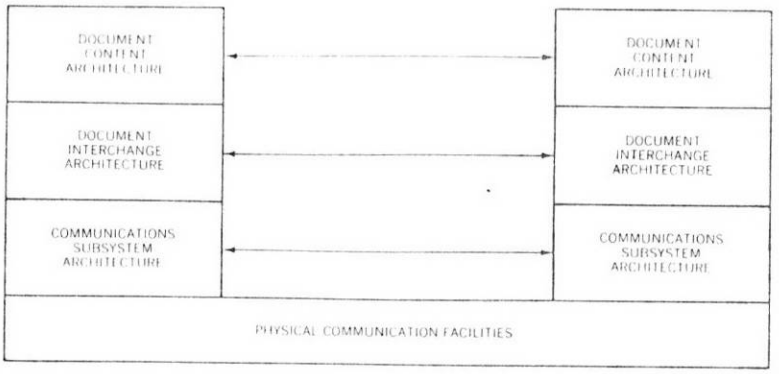
These elementary data objects become the components that form a page. A sample data stream is shown in Figure 9. Two elementary data objects are represented. Objects have a spatial relationship to the page in which they are contained but no relationship to each other.

The Final-Form Mixed Data Type DCA has three hierarchical levels: document, page, and elementary data object. Within this hierarchical scheme, control functions are applicable at particular levels. For example, the control function for media exists at a higher level than a page because multiple pages may appear on a single element of media.

As stated before, if the environment control information is explicitly specified within an object, the environment is restricted, or its "scope" is, to that object. If the environment control information is not specified, the environment for that object is taken from the next higher level in the hierarchy. This is referred to as "factoring."

To understand factoring, consider a mixed-data-type document in its simplest form, for example, a multiple-page document that is created by an inexpensive image scanner. All pages are identical in size, all pages contain one image with no other elementary data

Figure 10 DIA architectural layers



objects, and all images have identical data characteristics. Because the elementary data object environment control information for each image is identical, it can be specified once at the document level as opposed to once per image. In this example, the environment control information for all elementary data objects can be placed in a structured field called the *Master Environment Group*.

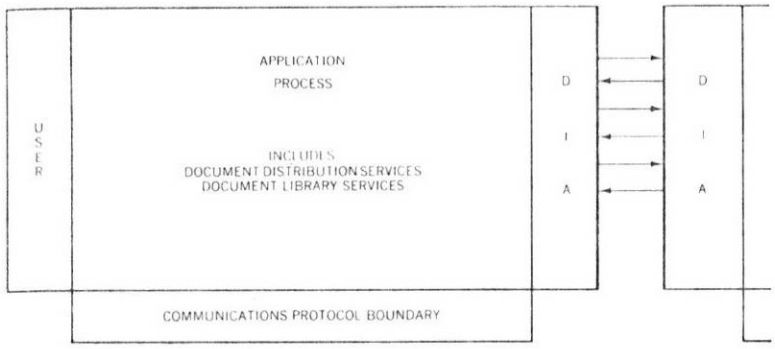
Every elementary data object that is within the scope of this Master Environment Group, but does not specify its own environment control information, uses the Master Environment Group as though the information had appeared in the elementary data object. Any elementary data object environment control information that appears within an elementary data object overrides the corresponding information in the Master Environment Group for the duration of that elementary data object. This maintains the integrity of the elementary data object as an independent entity.

The net effect is that the general case is allowed to appear at a higher level in the hierarchy. Any time it is necessary to override the general case, it can be accomplished by inserting the environment control information at the appropriate lower level in the hierarchy. Such environment control information remains in effect until the scope for that level of the hierarchy is reached.

Document Interchange Architecture

The Document Interchange Architecture (DIA) specifies how devices are to interchange intentions and data. It provides the capability to invoke the distribution services and the library services that are required of an interchange system. DIA specifies the rules and a data structure that establish the discipline for predictable information interchange between devices. DIA provides the method for systems and devices to interchange documents for a

Figure 11 DIA application process layer interfaces



variety of purposes, independent of the data types DIA contains. Interchange, as defined here, means that a document can be transported from one device to another device without change to its form or coherence. However, through the use of controls that can accompany the document, the document can be processed to produce the results defined by the sender.

DIA prescribes the exchange of information at the DCA layer and is distinct from the adjacent architectural layers: DCA and SNA (Figure 10).

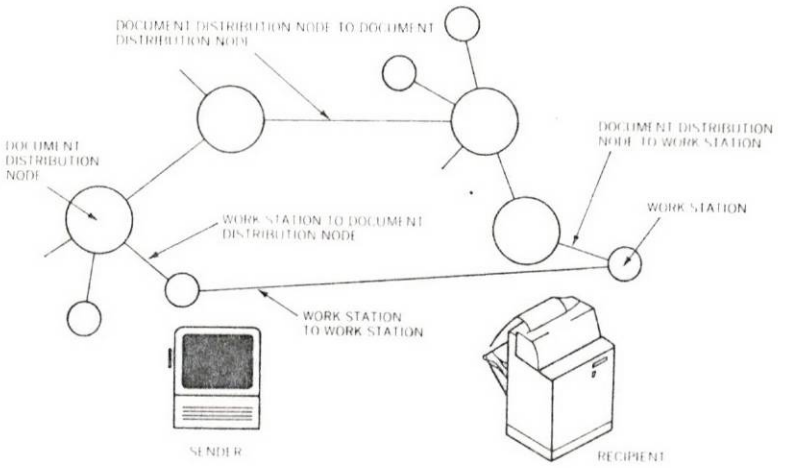
Expanding the DIA layer shows the relationship of DIA to the application process functioning at that layer. In Figure 11, the application process has an interface to the user, an interface to SNA, and an interface to another application process. The Communication Protocol Boundary provides the interface to SNA. DIA specifies the commands and the results expected on the interprocess interface; it does not address itself to either the user interface or the SNA interface. Data in DIA structures may physically pass through the Communication Protocol Boundary interface to reach the other process, but DIA defines the structure, commands, and results expected between the two application processes regardless of the physical path.

DIA is conceptually divided into an information-interchange base and various application services, as shown in Figure 12. The information-interchange base includes the structures and procedures that are common throughout the architecture (e.g., DIA session control, exception recovery, and encryption). DIA session control is the set of procedures and the commands necessary to exchange identification, authentication, functional capability, and status information that are pertinent to the application processes.

Figure 12 DIA structure

Document distribution services, illustrated in Figure 13, support document distribution through work station to document distribution node (DDN), DDN to DDN, and DDN to work station functions.

Figure 13 Document distribution services



Document distribution services also support work station to work station document interchange.

Document library services support the maintenance of documents on storage media. This facility provides the commands for requesting another process to perform operations, such as file, retrieve, and delete, on entire documents. This architecture does not address the manipulation of the internal content of a document.

The capability of the base and the parts may be subsetted to accommodate limited-function, entry-level devices that have communications facilities. This capability provides for orderly migration to more advanced devices.

The structure of DIA is extendable to other services of office automation systems.

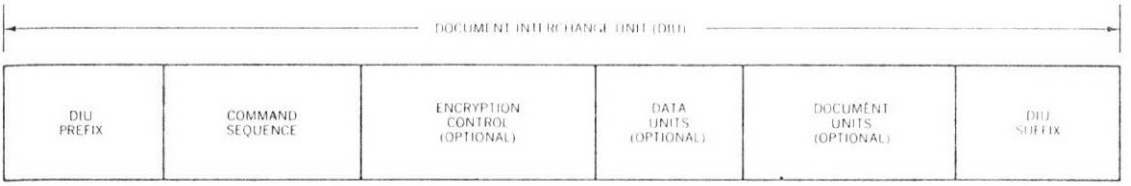
Document Interchange Unit

The Document Interchange Architecture defines a Document Interchange Unit (DIU), which is the major interchange facility between processes (Figure 14). An application process that uses DIA performs the construction of the DIU for sending to another process. The receiving application process interprets the structures of the DIU and performs the requested operations. The processing procedures are not defined by DIA; it is the responsibility of the application process to meet the requirements of the DIA interprocess interface and produce the specified results.

The DIU consists of six logical entities:

- The DIU Prefix introduces and identifies the DIU.
- The Command Sequence contains the commands for application function processing.

Figure 14 Document Interchange Unit



- The Encryption Control contains the information to encipher the data units and document units for security purposes.
- The Data Unit contains information that may be referenced by one or more commands in the Command Sequence.
- The Document Unit contains the document profile, which describes the characteristics of the document and, optionally, the content of the document.
- The DIU Suffix specifies the end of the DIU and indicates whether or not any abnormal conditions affected the DIU transmission.

The DIU, in this information interchange context, is assumed to carry documents that conform to the DCAs discussed earlier. However, the DIU, in general, can transport any type or format of the data object. The DIU carries information that explicitly identifies the object type and the characteristics necessary to process it as the sender intended. If an information interchange object does not conform to the DCAs described, a transformation may be required.

Each DIU component and subcomponent has an introducer that specifies the length and describes the semantics and syntax of that piece of data. Figure 15 shows an overall perspective of the DIU structure.

Again for further insight, the Request Distribution Command, which is used by a work station to request the distribution of a document, is shown in Figure 16. The command is schematically depicted within a DIU.

Concluding remarks

The following scenario is intended to put the various architectural components into perspective and to show how an office-information-interchange system might work. An actual working system could be considerably different from what is outlined here because of office procedures, product mix, and applications to be performed.

Figure 15 DIU overview

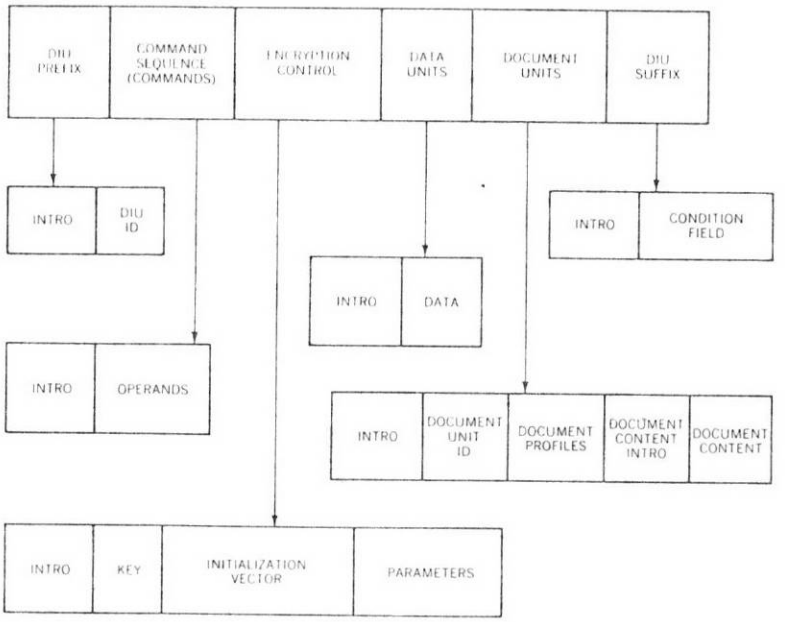


Figure 16 The Request Distribution Command

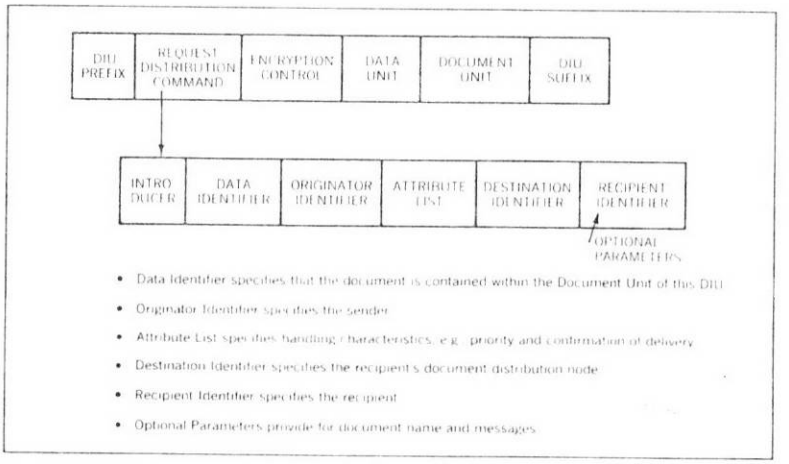


Figure 17 depicts the initial phase of information interchange—document creation. The numbered steps show the order of the operations. A document is started using the revisable-form data stream at Work Station A (Step 1). The document is moved to Work Station B before it is finished (Step 2). It can be moved via communications facilities or diskette. How it is moved will probably be influenced by the physical location of the work stations.

Figure 17 Initial phase of information interchange

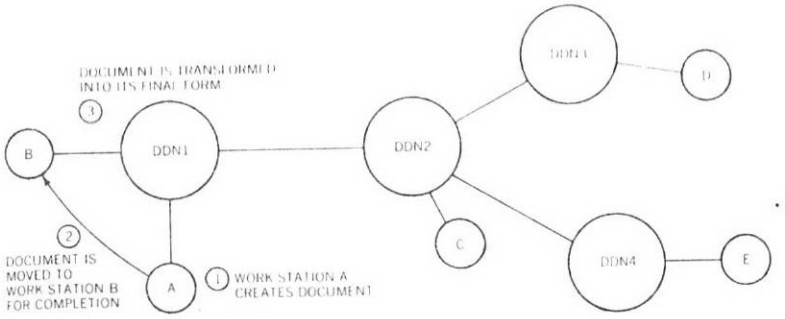
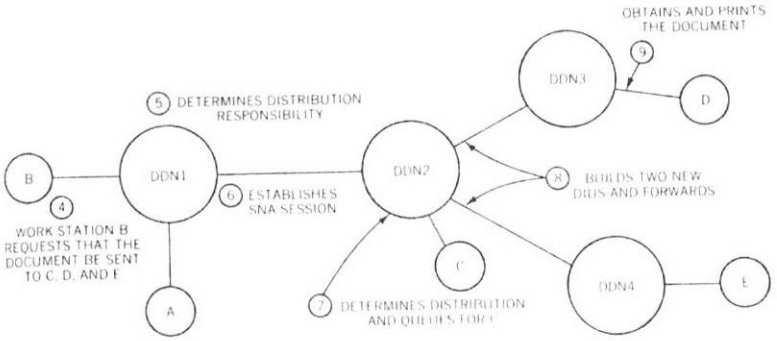


Figure 18 Document distribution information interchange



The two work stations need not be identical devices. After being moved, the document is completed and the data stream is transformed into its final form (Step 3).

Figure 18 depicts the document distribution application. Work Station B requests DDN1 (Document Distribution Node 1) to distribute the document to Work Stations C, D, and E (Step 4). DDN1 determines its distribution responsibility by examining the addresses contained in the request distribution DIU and accessing its destination directory (Step 5). DDN1 established an SNA session with DDN2. The two nodes exchange DIA session control information, and the document is sent to DDN2 (Step 6). DDN2, which has stored the document, determines its distribution responsibility. Because Work Station C is a subscriber at DDN2, DDN2 builds a delivery DIU and queues the DIU until a session is established with Work Station C (Step 7). The destination directory shows that Work Station D must be routed through DDN3 and Work Station E through DDN4. DDN2 builds two new distribute DIUs and forwards them or stores them for later transmission (Step 8). Finally, Work Stations D and E obtain the document from their controlling

nodes and print or display it (Step 9). This simplified presentation demonstrates one way in which the system could work.

summary What has been described is a set of architectures, some already implemented in existing products, that in combination with other IBM architectures will permit information interchange across a broad spectrum of user applications and office automation offerings. Initially, information interchange will be restricted to text with a limited number of devices, but over the next few years, information interchange is expected to become available across a wide offering of systems. It is also expected that data types other than text will be included, over time, for interchange.

ACKNOWLEDGMENT

The architecture developments referred to in this paper were conceived and developed by many contributors who represented several IBM products and development locations. The number of contributors is too large to list them individually, but if it had not been for the dedicated work of these people, this paper would not have been possible.

CITED REFERENCES

1. C. R. Blair and J. P. Gray, "IBM's Systems Network Architecture," *Datamation* **21**, No. 4, 51–56 (April 1975).
2. J. H. McFadyen, "Systems Network Architecture: An overview," *IBM Systems Journal* **15**, No. 1, 4–23 (1976).
3. P. G. Cullum, "The transmission subsystem in Systems Network Architecture," *IBM Systems Journal* **15**, No. 1, 24–38 (1976).
4. J. P. Gray and T. B. McNeill, "SNA multiple-system networking," *IBM Systems Journal* **18**, No. 2, 263–297 (1979).
5. *Systems Network Architecture Format and Protocol Reference Manual: Architecture Logic*, SC30-3112, IBM Corporation; available through IBM branch offices.

GENERAL REFERENCES

J. N. Bruno, "Electronic mail: It gets there fast," *Administrative Management*, 28–70 (September 1979). Discusses the general concept of electronic mail with its inherent product-type and data-form problem.

Dun's Review (August 1979). The entire issue is devoted to the "Office of the Future."

G. H. Engel, J. Groppuso, R. A. Lowenstein, and W. G. Traub, "An office communications system," *IBM Systems Journal* **18**, No. 3, 402–431 (1979). Discusses the applications requirements of an office communications system.

IBM Administrative System, Introduction, GC23-0702, IBM Corporation (September 1979); available through IBM branch offices.

IBM Distributed Office Communications System: Host Programmer's Guide, GA33-3030, IBM Corporation; available through IBM branch offices.

The author is located at the IBM Information Systems Division laboratory, P.O. Box 1900, Boulder, CO 80302.

Reprint Order No. G321-5138.

N-44

Development of the GILT message standard - a progress report by the GILT Message group

ABSTRACT

The GILT message standard makes possible the transfer of messages between independent computer-based message systems (CBMS). The independent CBMSs are interconnected via X25 data networks. The GILT standard is developed within the framework of ISO Open Systems Architecture reference model and is described in terms of an "open CBMS" (OCBMS). The functional classification of OCBMSs is hierarchical, and a higher class OCBMS contains all the functions of all lower OCBMS classes. The GILT standard results in the specification of an OCBMS/OCBMS message exchange protocol, where the message is described in terms of globally significant attributes and is in this sense a virtual message. Further work is needed in the areas of detailed protocol specification, naming strategy, interaction of different-class OCBMSs, and OCBMS description in Open System terms.

CONTENTS

1. The purpose of the GILT message standard

GILT is an international cooperative research project whose aim is to devise practical methods of interconnecting independent computer-based message systems (CBMS) using existing networking techniques. The overall GILT structure is described elsewhere. This paper describes the work, achievements and future goals of the Message subgroup of GILT.

The aim of the Message group is to define standard procedures to permit the exchange of messages between existing CBMSs. Such procedures will permit users of distinct systems to exchange messages without knowing the nature or operating conventions of the remote system.

2. The GILT approach, and some terms defined

For GILT purposes, a CBMS is a store-and-retrieve text message switch. Different CBMSs may vary greatly in capability, ranging from a standard Teletex terminal through conventional mail systems and up to advanced conferencing facilities. But the essential feature of a CBMS is that it supports one or more local mailboxes.

A mailbox is an abstraction of "enduser" and is the source or destination of a message. The correspondence between mailbox and enduser is not necessarily 1:1, and is of purely local significance.

The GILT approach assumes the existence of a number of independent CBMSs, each with access to an X25 data network supplied by the national PTT. The national networks are interconnected by X75 gateways and form a fully-connected CBMS network : every CBMS is directly connected to every other CBMS and no relaying via an intermediate is required.

Although the independent CBMSs are now physically connected and can exchange data, they have no common understanding of how data from another CBMS should be interpreted. The GILT Message group supplies this common understanding by defining a standard for CBMS/CBMS message exchange. The GILT message standard consists of a virtual message definition (in terms of globally significant attributes) and a set of service primitives for operating on virtual messages.

The virtual message attributes are globally significant in the sense that they are abstract message components whose meaning is independent of their implementation in any given CBMS. The mapping between global and local attributes is not the concern of GILT, except to the extent that the standard is devised so as to minimise the conversion difficulties for a number of CBMSs which are believed to be representative.

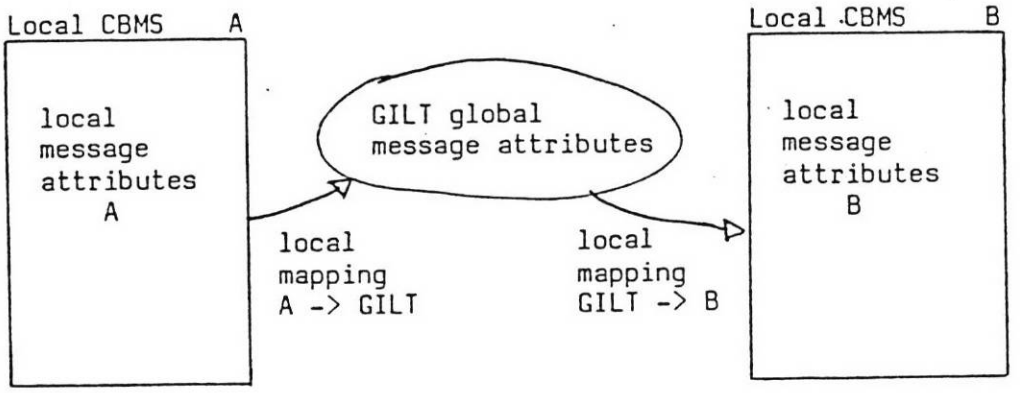


Figure 2.1 The GILT message standard is a CBMS/CBMS message exchange protocol. Independent CBMSs exchange messages in terms of globally significant attributes. The local/global mapping of attributes is not part of the GILT standard, and is of purely local concern.

The service primitives are chosen to provide the basic function of CBMS-to-CBMS message exchange, but not necessarily to express the full message-handling functionality of any given CBMS. As with virtual message attributes, the correspondence between GILT functions and local CBMS functions is of purely local interest and is not the concern of GILT.

The net effect of two or more independent CBMSs adopting the GILT message standard is that they have available a CBMS/CBMS message exchange protocol. A message originating at a mailbox in one CBMS may be delivered to a mailbox on another CBMS.

3. GILT architecture

All GILT work proceeds within the framework of the ISO Open Systems Architecture reference model (reference 1). The aim of GILT is to define in terms of ISO levels 5, 6 and 7 the communication needs of interconnected CBMS applications. Level 5 is the province of the GILT session group and is described elsewhere. The Message group considers levels 6 and 7.

The details of the upper layers of the reference model are yet to be fully elaborated. It is recognised that the process of elaboration will benefit from a consideration of specific intercommunication requirements. GILT expects to contribute to ISO understanding as a result of practical experience gained from implementation.

Few CBMSs are built with interconnection in mind and even fewer in accordance with Open Systems Architecture. A particular CBMS therefore serves a set of mailboxes which may be termed local. The CBMS to which a mailbox is local is its home system. At present there is no generally agreed method of communication between mailboxes which have different home systems. A functional entity is required to enable one CBMS to communicate with

another in an Open Systems environment. This entity may be called an open CBMS (OCBMS).

Interconnecting of computer-based message systems (CBMS) in a standardised way means adding the seven-layer ISO hierarchy to an existing CBMS. This will make the local CBMS an OCBMS. The set of services provided by the ISO hierarchy will define a functional window which specifies a standardised or virtual OCBMS. The OCBMS window makes the facilities of the local CBMS available to other OCBMSs interconnected via the underlying communication network. The functionality of this window will usually be less than that available in the local context of that CBMS.

There are two important points to note about the OCBMS concept. First, it is the OCBMS which is the subject of standardisation and not the local CBMS. The mapping between the OCBMS functions and the local CBMS services is of purely local concern. If this mapping is difficult for the full range of functions then a particular CBMS may choose a subset and thus implement a narrower OCBMS window. However, useful interworking of OCBMSs requires a non-trivial overlap of functions. The second point to note is that the entities in the ISO hierarchy are concerned only with communication. Thus, the OCBMS is a protocol driver for the message exchange protocol, and imposes no implementation constraints on the local CBMS.

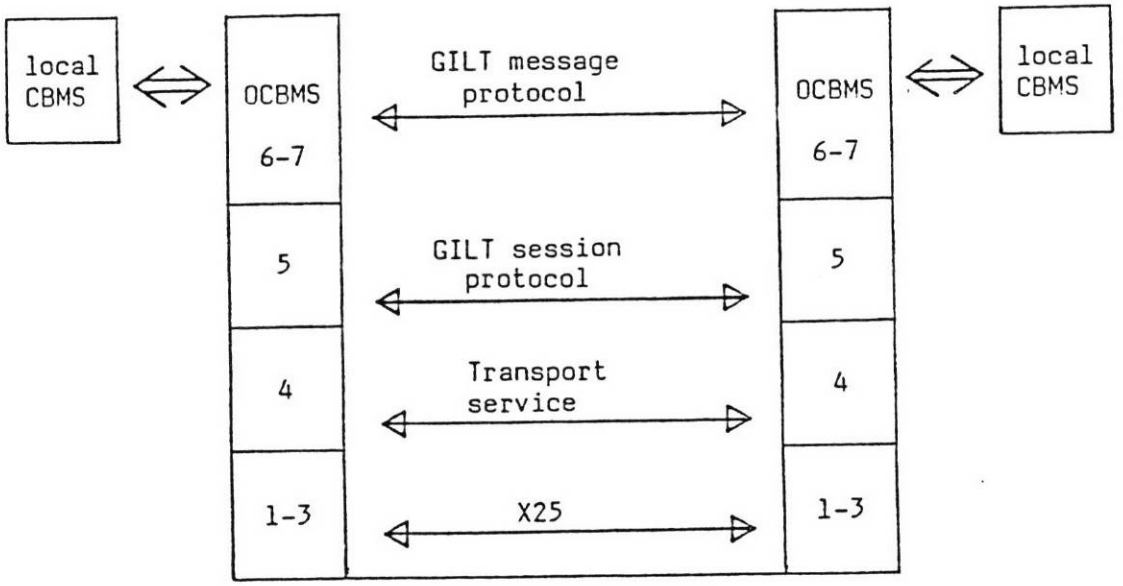


Figure 3.1 CBMSs communicate by implementing a functional window onto the ISO Open Systems environment. The window is called an OCBMS and its standardisation yields the GILT message exchange protocol.

4. The GILT OCBMS

4.1 Functional classification of OCBMSs

A particular CBMS does not necessarily make all local functions available to a remote CBMS. The OCBMS functionality is usually more restricted than the local functionality: a CBMS may have very sophisticated local functions but offer a rudimentary open service. Further, different CBMSs may choose to make available different OCBMS functions. Meaningful interaction between interconnected CBMSs is only possible if they have commensurable OCBMS functionality. To achieve this, GILT classifies OCBMS functions into globally known categories, which are hierarchical in the sense that all functions of class N-1 are contained in class N. Again, note that GILT is concerned with OCBMS rather than CBMS functionality.

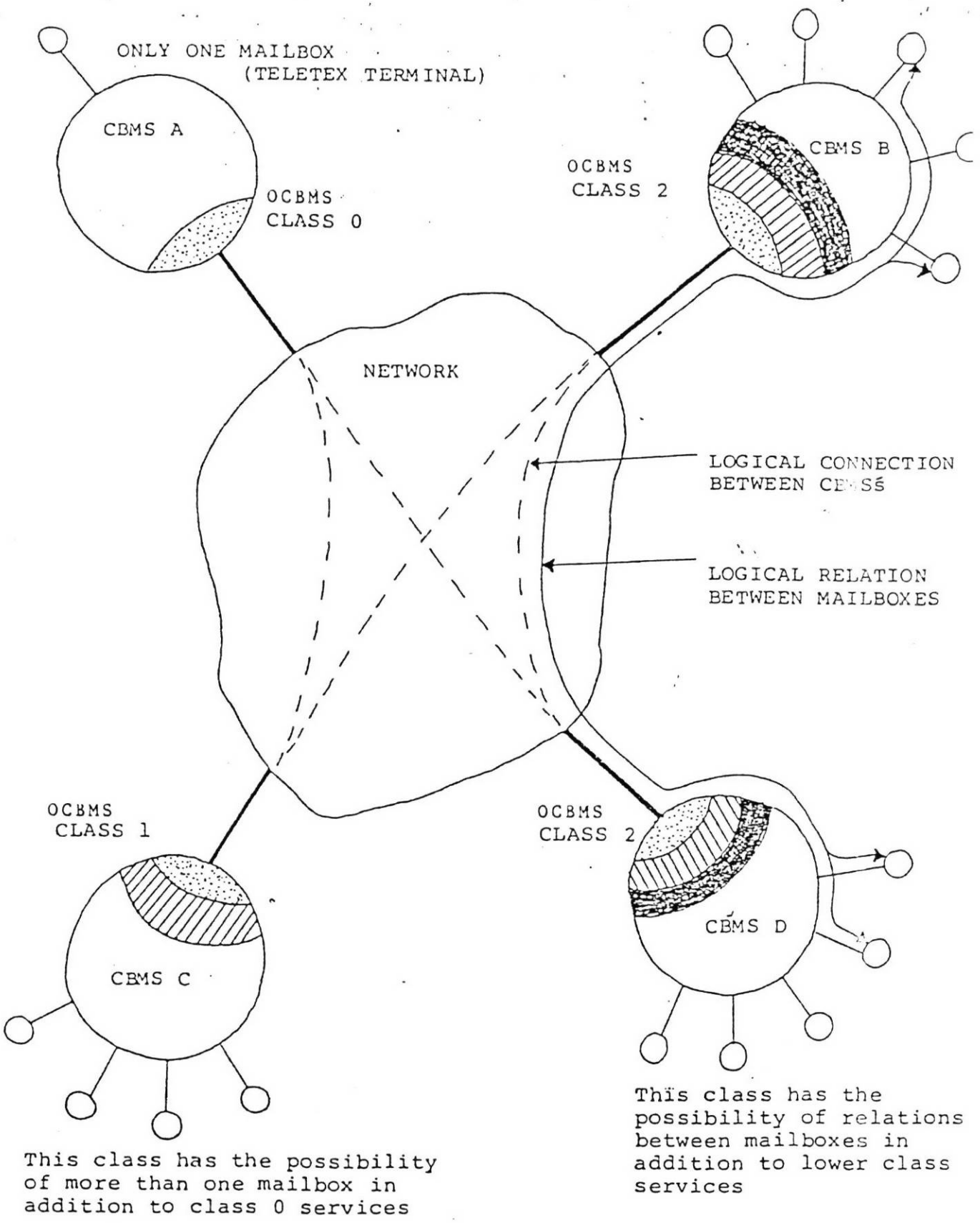
GILT has defined 3 OCBMS classes, designated Ø, 1 and 2. These classes are distinguished on two criteria: the number of mailboxes supported and the open services offered.

Class Ø corresponds to the Teletex service. (The significance of this choice is that GILT bases its work on existing standards which are both stable and of economic importance. Teletex (reference 2) fulfils these criteria). The OCBMS supports a single mailbox which, at any one time, can send to or receive from exactly one mailbox in another CBMS.

The class 1 OCBMS contains all the functionality of class Ø. In addition, it supports one or more local mailboxes and can obtain information concerning mailboxes in other OCBMSs.

The class 2 OCBMS contains all the functionality of classes Ø and 1. In addition class 2 is capable of maintaining a logical relation which links together a group of mailboxes. This relation is interpreted (in some way) as a joint ownership of messages and, for example, includes the concepts of a "conference" where a message from one mailbox is distributed to all participating mailboxes. Other interpretations of the relation concept are possible. The establishment, control and mechanisation of class 2 relations in an Open System environment (and the extent to which these considerations should be part of the GILT standard) are currently being studied by GILT.

This OCBMS class structure is described in reference 4.



Interconnection of CBMS's by adding ISO open systems functions and classifying those functions

4.2 OCBMS service primitives

On behalf of the local CBMSs each OCBMS offers open services to remote OCBMSs and obtains similar services from remote OCBMSs. These services are invoked by primitives of the GILT message protocol, and five such service primitives have been identified. These primitives are sufficiently general purpose for all GILT applications considered to date. They are:-

send	send message
read	read message
relation	create relation delete relation add participant remove participant
list	list mailboxes list mailbox description list relations list relation description
control	abort status delivered confirmation action (e.g. read) confirmation undeliverable mail

The "send" service enables a CBMS to transmit a message to another CBMS. The "read" service enables a CBMS to request a message from another CBMS. The "relation" service enables a CBMS to establish and manage relations in another CBMS (e.g. start up a conference). The "list" service obtains descriptive information on the mailboxes and relations of another CBMS. The "control" service obtains monitoring information on transfer activities in progress.

The parameter fields and their default values of these service requests are still under discussion.

A class Ø OCBMS (i.e. a Teletex terminal) contains only the "send" function. A class 1 OCBMS contains the additional functions "read", "list" and "control". A class 2 OCBMS contains all five functions.

4.3 Global message attributes

For message transmission purposes, the GILT message standard defines a number of globally significant attributes of a message. All message transfers between OCBMSs are in terms of these global attributes, which thus constitute the first step towards a virtual message definition. The GILT standard in this area owes much to a National Bureau of Standards report (reference 3) which surveyed the characteristics of a representative sample of existing CBMSs. GILT has discussed the NBS conclusions at length and produced a modified list of message attributes

suitable for the identified GILT purposes. Some of these attributes are listed below. It should be emphasised that GILT discussion on this matter is not yet complete. A detailed description of all message attributes considered to date can be found in reference 5.

attribute group	attributes
originator	from copyright forwarded by
recipient	to distribute to do not distribute to
date	date posted validity
message handling	message identity character set restrictions take action on send reply to
message content	subject abstract keywords text
cross-reference	in reply to comment on revision of

As well as defining message attributes, it is necessary to agree on some representation of these attributes and their contents. GILT has adopted the so-called GILTONE standard representation which is adopted with Teletex compatibility in mind. This consists of all printable characters of the ISO 646 alphabet, plus colon, space, linefeed and formfeed. Both linefeed and formfeed imply carriage return. All linefeeds immediately preceding a formfeed are ignored. For recognition purposes, upper and lower case characters are equivalent. The maximum length of a line is 77 characters.

5. Directions of future work

Further work is needed on the GILT message standard in order to achieve its purpose of permitting the exchange of messages by independent CBMSs. The semantics of the virtual message attributes and of the CCBMS service primitives needs refining. This will result in a more precise syntax of the service primitives, and the specification of the standard in terms of a message exchange protocol.

Underlying this work are a number of problem areas which are difficult because the problems are not yet well understood. As a result it is difficult to formulate precise questions, and suggested solutions reflect personal preference to a large extent.

One such area is the naming strategy used to direct messages from one mailbox to another. There is a wide range of naming conventions in use by existing CBMSs. The GILT solution must accommodate the conventions adopted by all project participants, but should in some sense be generally applicable rather than merely a compromise.

Another area is the handling of functional asymmetry that arises from the hierarchical definition of GILT CBMS classes. A lower class N OCBMS should have access to the appropriate subset of functions of a higher class M OCBMS. In practice it is difficult to foresee the implications of this asymmetric interaction. This has a bearing on the GILT functional models of OCBMSs, and on the logical structure of the message exchange protocol.

Yet another area is the position of various GILT functions within the ISO reference model. GILT expects eventually to arrive at a reasonable satisfying answer, but takes the pragmatic view that insight will grow as work proceeds.

6. References

1. Provisional reference model of Open Systems Architecture, ISO/TC97/SC16/WG1/N74

2. Final CCITT report to the VIIIth plenary assembly (Part III) - new and revised recommendations, CCITT Study Group VIII, June 1980

3. Specification of a draft message format standard, NBS Report ICST/CBOS-80-2, October 1980, Bolt Beranek and Newman Inc.

4. A proposed standard for interconnection computer-based message systems, internal working paper, GILT/MES/UNINETT/004

5. GILT message structure standard, internal working paper, GILT/MES/FOA/005

APPENDIX 1 : definition of GILT terms

CBMS	a CBMS is a store-and-retrieve text message switch. The essential feature of a CBMS is that it supports one or more mailboxes.
mailbox	a mailbox is an abstraction of the enduser (human or otherwise) of the CBMS, and is the source or destination of a message. The correspondence between endusers and mailboxes in a CBMS is not necessarily 1:1.
OCBMS	the functional window which interfaces a CBMS to the ISO Open Systems environment. The OCBMS is concerned exclusively with CBMS/CBMS communication and imposes no implementation constraints on the local CBMS.
relation	a logical association of mailboxes of different CBMSs which (in some sense) is interpreted as joint ownership of mail. A teleconference is an example of a relation.
actor	a human (always!) source of authority who can authorise CBMS activities, accept liability for actions, sign contracts, etc. Actors may be organised into a hierarchy.
document, message	a document is static information in a mailbox. A message is information in transit between mailboxes. (This distinction need not be made explicitly where there is no danger of ambiguity).

DIALOG INFO

Oct 1985

DIALMAIL Basics

October 1985

Terms and Conditions
Applicable to the DIALMAIL℠ Service
Effective October 1, 1985

DIALOG Information Services, Inc. (DIALOG) hereby publishes the following terms and conditions for Customer usage of the DIALMAIL℠ Service, effective as of the above date. DIALOG reserves the right to modify these terms and conditions at any time, effective upon publication. Customer's use of the DIALMAIL Service constitutes agreement to these terms and conditions and any modifications thereof.

1. Customer agrees that the following uses of the DIALMAIL Service are strictly prohibited. Customer agrees to indemnify, hold harmless and defend DIALOG from and against any and all claims, actions, suits, judgements, and expenses (including court costs and reasonable fees of attornies, accountants, and expert witnesses), at Customer's sole expense, arising from Customer's failure to abide by these restrictions on use of the DIALMAIL Service.
   A. Use of the DIALMAIL Service for wire transfers of funds;
   B. Unauthorized communication of any charge or credit card information belonging to any other person or entity;
   C. Unauthorized communication of any information concerning any password or other online access number, code, or identification or any other proprietary information belonging to any other person or entity;
   D. Use of the DIALMAIL Service to copy or to distribute or transmit copies of copyrighted materials belonging to any other person or entity without the permission of the copyright owner(s).
   E. Communicating any obscene or defamatory information including but not limited to on bulletin boards or in conjunction with conferences; or
   F. Use of the DIALMAIL Service in violation of any telecommunication, postal, or other local laws or regulations of the Customer's country of origin or of the United States or in furtherance or in the commission of any crime or other unlawful or improper purpose.

2. DIALOG AND ITS SUPPLIERS MAKE NO WARRANTIES OR REPRESENTATIONS OF ANY KIND WITH RESPECT TO THE DIALMAIL SERVICE, WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO MERCHANTABILITY OR FITNESS FOR PARTICULAR PURPOSE AND NEITHER DIALOG NOR ITS SUPPLIERS ASSUME ANY RESPONSIBILITIES WITH RESPECT TO CUSTOMER'S USE THEREOF. IN NO EVENT WILL DIALOG OR ITS SUPPLIERS BE LIABLE FOR INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. IN NO EVENT SHALL DIALOG'S LIABILITY FOR DAMAGES, REGARDLESS OF CAUSE OR FORM OF ACTION, WHETHER IN CONTRACT OR IN TORT (INCLUDING NEGLIGENCE, DEFAMATION, AND/OR PRIVACY ACTIONS), EXCEED THE GREATER OF (1) THE INVOICE VALUE OF CUSTOMER'S DIALMAIL SERVICE USAGE FOR THE MONTH DURING WHICH THE CAUSE OF ACTION ACCRUED, OR (2) ONE HUNDRED DOLLARS (U.S.$100). THE FOREGOING CONSTITUTES DIALOG'S SOLE AND EXCLUSIVE LIABILITY TO CUSTOMER WITH RESPECT TO THE CUSTOMER'S USE OF THE DIALMAIL SERVICE.

3. Customer further agrees:
   A. Transmission of confidential business and sensitive personal information is at Customer's sole risk;
   B. DIALOG reserves the right to monitor and review transmissions online and in storage, and to remove or reject any material which DIALOG, at its sole discretion, believes may be unlawful or objectionable, without prior notice to Customer;
   C. Except as otherwise expressly set forth herein, all Terms and Conditions governing Customer's use of the DIALOG Information Retrieval Service and/or the KNOWLEDGE INDEX℠ Service, whichever is applicable, are applicable to Customer's use of the DIALMAIL Service.
   D. In the event that Customer is delinquent in paying invoices for charges incurred, DIALOG may remove data stored without prior notice.

THE MAIN MENU

Every time you connect to DIALMAIL, you will be greeted by the Main Menu:

```
DIALMAIL  -  21SEP85 13:05 EST.
You have 2 items in your Inbox.

You may enter:
   SCAN      to list your messages
   READ      to see your messages
   CREATE    to prepare a message
   PROFILE   to change user profile
   HELP      for assistance at any time
   MENU      to return to the main menu at any time
   EXIT      to leave DIALMAIL
Your command:
```

The first three options (SCAN, READ, and CREATE) are the commands you will use to send and receive mail. They are described in further detail on the following pages.

The option PROFILE enables you to change your DIALMAIL user profile.

The last three options on the menu (HELP, MENU, EXIT) are used to request a HELP message, to return to the Main Menu, or to exit from DIALMAIL. The commands EXIT (entered in full) and LOGOFF may be used at any time to disconnect your terminal from DIALMAIL.

Once you become familiar with the various menus on DIALMAIL, you may opt for SHORT menus or NONE by entering the appropriate word at any time, or by changing the "Size of Menu" option in your Profile (see p.19). To return to the longer default menus, type in LONG at any time.

```
Your command: short

DIALMAIL  -  21SEP85 19:32 EST

Scan  Read  Create  Profile  Help  Menu  Exit

Your command:
```

All menus are presented in their LONG format the first time they appear in this guide. Thereafter, they appear in their SHORT format.

HELPFUL HINT: Most options on any menu may be abbreviated to their first letter, e.g., SCAN may be entered as S. Some exceptions are COPY (CO), FORWARD (FO), and FORM (FORM). **A space is required after a command or its abbreviation.**

How To: Read Mail

Mail is placed in your Inbox when it arrives. You are told how many messages there are when you first connect to DIALMAIL (see Main Menu, p.1). From the Main Menu, you may enter either SCAN or READ to receive a menu of choices; from this menu you may choose INBOX:

```
You may enter:
   INBOX      to read your unread messages
   DESK       to read messages you have read today
   FOLDER     to read messages within a folder
   BBOARD     to read messages within a bboard
   CONF       to read messages within a conference
   MENU       to return to the main menu
```

When using SCAN followed by INBOX, you will receive a response similar to the following, with the messages listed in the same order in which they were received (i.e., the oldest message first):

```
2 messages in Inbox

No.    Posted      Lines  From                Subject              Type
  1 SEP18 16:25       36 Barbara Sievers   User group meeting       Msg
  2 SEP19 19:45       17 Norm Sullivan     Report draft             Msg

You may enter:
    READ       to read the messages you scanned
    DELETE     to delete messages you scanned
    PRINT      to print messages you scanned
    SCAN       to scan for other messages
    HELP       for assistance
    MENU       to return to the main menu

Your  command:
```

When using READ from the Main Menu followed by INBOX, you will receive the complete text(s) of your message(s), one at a time, in the order in which they were sent.

HELPFUL HINT: To directly scan or read your mail, simply enter SCAN INBOX (or S I) or READ INBOX (or R I) from the Main Menu, and skip the intermediate menu!

When using READ from the Main Menu followed by INBOX, you will receive the complete text(s) of your message(s), one at a time, in the order in which they were received. This is shown in the example on the following page:

```
Your command: read inbox
3 messages found in inbox

Date:     Monday Sept 09, 1985 15:43   EST      Msg ID: 3309
From:     Tim Hammond

To:       Jan Wright
Subject:  DIALMAIL

Glad to see you are using DIALMAIL. I think you will find it
very useful, especially for keeping up to date with committee
work, and for communicating effectively with colleagues.  Your
DIALMAIL Inbox will receive and store messages for you until  you
are ready to view them.  The conferences and bulletin boards
offer valuable ways to keep in touch as well.  Why don't you send
me next week's meeting agenda on DIALMAIL?  I'll check for it on
Friday.
Regards, Tim

You may enter:

    READ       to read this message again
    ANS        to answer the sender of this message
    DELETE     to delete the message
    FILE       message in a folder
    FORWARD    to forward message to other users
    EDIT       to edit a copy of the message
    NEXT       read next message
    MENU       return to the main menu

Your command:
```

The menu shown above offers the following options:

READ repeats the sequence shown above.
ANSWER addresses an answer and prompts you for text.
DELETE completely erases the message you just read.
FILE places the message in the Folder you designated.
FORWARD creates a copy of the message to send on.
EDIT creates a copy of the message for you to edit.
NEXT places the message on your Desk and READs the next message.
MENU places the message on your Desk and returns you to the Main Menu.

HELPFUL HINT: To send a message from a Folder, Conference, or Bulletin Board to someone else, READ the message. This will generate the menu shown above, allowing you to either FORWARD or EDIT it.

How To: Send Mail

You send mail out through the CREATE option on the Main Menu. Enter MESSAGE (or M) from the subsequent menu. The choice of this option automatically generates the TO: prompt, following which you simply enter the last name of the person to whom you wish to send the message. **The TO: prompt repeats until you enter a simple (RETURN).** You are then prompted for a subject line, followed by the text of your message. You may also enter CREATE MESSAGE (or C M) directly from the Main Menu:

```
You may enter:
Scan  Read  Create  Profile  Help  Menu  Exit

Your command: create message
Enter message address or ‹return› to end addressing
To: sullivan
Message addressed to: Norm Sullivan
To: (RETURN)

Subject (up to 40 char): Comments on report draft

Enter text, ending it by typing a period on a blank line.

?
```

Enter the text of your message in lines no longer than 80 characters. You must end the text by entering a period on the line following the end of your message. This will produce the following menu:

```
You may enter:
    READ       to review your message
    EDIT       to edit the text or ENVelope of your message
    SEND       to send your message
    CANCEL     to cancel the message
    FILE       in a folder and do not send
    MENU       to return to the main menu

Your command:
```

The EDIT option will allow you to make changes to either the text of your message or the ENVelope, i.e. the addresses to which you wish to send the message. See HOW TO: Edit a Message, p.7.

HELPFUL HINT: The MENU option listed above will not only return you to the Main Menu but will also automatically cancel the message you were working on.

You may send copies of messages to other DIALMAIL users as well as to Folders, Mailing Lists, Conferences, or Bulletin Boards through the use of the following 'tags:'

CC (courtesy copy sent to a person),
BCC (blind courtesy copy sent to a person),
F (copy sent to a folder),
L (copy sent to a mailing),
C (copy sent to a conference), or
B (copy sent to bulletin board)

In response to the TO: prompt, simply indicate the appropriate code in front of the addressee's name:

```
TO: cc wallace
  Message addressed to: Maureen Wallace
TO: bcc steiner
  Message addressed to: Paula Steiner
TO: f user group news
  Message addressed to file folder USER GROUP NEWS
TO: (RETURN)
```

Messages may also be tagged for:

ANS (answer),
DEL (delay sending message),
NOF (no forwarding allowed),
PAP (send paper copy in U.S. mail - see p.10),
REA (readable once by addressee),
REC (return receipt), or
URG (urgent)

by entering one or more of these parameters following a left parenthesis after the person's name:

```
TO: wallace (rec urg
  Message addressed to Maureen Wallace
TO: cc steiner (nof
 Message addressed to Paula Steiner
TO: (RETURN)
```

HELPFUL HINT: You may stack multiple addresses by placing a semicolon after each name:

```
TO: wallace (rec; cc steiner; f user group news
  Mesage addressed to: Maureen Wallace
  Message addressed to: Paul Steiner
  Message addressed to file folder USER GROUP NEWS
```

How To: Create A Mailing List

You may establish permanent Mailing Lists for groups to whom you regularly send messages. This is done using the CREATE option on the Main Menu followed by the word LIST (or simply C L), as shown below:

```
You may enter:
Scan Read Create Profile Help Menu Exit

Your command: create list
Creating Mailing List

Please type a short title (up to 40 char) for the mailing list.

Title: jogging

Please enter the names in the mailing list.

Enter name: andrews
Richard Andrews is now on the mailing list.

Enter name: wallace
Maureen Wallace is now on the mailing list.

Enter name: hammond
Tim S. Hammond is now on the mailing list.

Enter name: (RETURN)
```

The ENTER NAME: prompt will continue to appear until you enter a simple (RETURN). At this point the Mailing List has been created, and you will be returned to the Main Menu.

HELPFUL HINT: To send mail to your Mailing List, use the CREATE MESSAGE option from the Main Menu, and in response to the TO: prompt, enter the name of your list preceded by an L (for LIST), e.g.,

```
TO: l jogging

Message addressed to: Mailing list JOGGING
TO: (RETURN)

Subject (up to 40 char.):
```

How To: Edit A Message

After you finish entering the text of your message by placing a period on a blank line, you may EDIT the text. When you choose the EDIT option from the menu (shown on p.3), you will be told the number of lines in your message. Lines are numbered in units of 10, so that you may insert new lines (e.g., INSERT line 45 between lines 40 and 50). You will then receive a menu of options for viewing and editing your message:

```
There are 5 lines. The last line number is 50.
You may enter:
SAVE      - to save and exit file
LIST      - to see your current message
HELP      - or ? for detailed EDIT commands
INSERT    - to add text to the end of your message

EDIT: Enter command or SAVE to exit
```

There are nine EDIT commands available for your use. Most commands are entered followed by a space and the number or number range of the line(s) you wish to change. The commands and their formats are listed below ("nn" indicates a line number):

CHANGE - To change text on a given line, enter CHANGE (or C) followed by the line number, a slash, the character string to be changed, a slash, and the correction:

```
? change 40/at that time/for a review
  40 proposal to us for a review? Thanks.
  One line changed.
```

COPY - To copy line(s) to another position in the message, enter COPY nn TO nn. The line(s) indicated will be duplicated to the new position. May be abbreviated to CO, e.g., CO 20-40 TO 115

DELETE - To delete line(s), enter DELETE nn or DELETE nn-nn. May be abbreviated to D, e.g., D 20-90

INSERT - To insert new line(s) between two existing lines, enter INSERT (or I) followed by the number of the first new line you wish to add. You will be prompted for subsequent lines until you enter a period:

```
? i 45
INSERT: (end by typing a period on a blank line):
   45
? Let me know if you can't make it.
 46
? .
EDIT: Enter command or SAVE to exit
```

To add new lines at the bottom of the message, simply enter INSERT (or I) followed by a (RETURN). You will be prompted for additional lines of text.

LIST - To view line(s) of your message, enter LIST nn or LIST nn-nn. The command LIST entered alone will display all lines (LIST ALL may also be used). May be abbreviated to L, e.g., L 20-90.

```
? list all
   10 Nathan,
   20 The budget committee will meet tomorrow at 9
   30 in my office. Could you bring your project
   40 proposal to us for a review? Thanks.
   45 Let me know if you can't make it.
   50 Christine
```

To locate a line containing certain TEXT, enter LIST /TEXT/ (or L /TEXT/):

```
? l /review/
40    proposal to us for a review? Thanks.
```

MOVE - To move line(s) within the text, enter MOVE nn TO nn. May be abbreviated to M, e.g., M 110-160 TO 280

QUERY - Shows total number of lines in the text and the number of the last line.

RENUM - Renumbers the lines of the text in units of ten. To specify units of other than ten, enter a different number, e.g., R 5 will renumber in units of five (5, 10, 15, 20, etc.).

SAVE - Ends your editing session, stores your results, and returns you to the previous list of options (see p.3):

```
Read  Edit  Send  Cancel  File  Menu
```

HELPFUL HINT: You may also edit a copy of a message stored in a Folder, Conference, or Bulletin Board after you have READ it. The original message will remain where it is stored until it is DELETEd. The edited copy may be sent to another user, another Folder, etc.

How To: Edit An Envelope

You may EDIT the envelope (i.e., the addresses) of a message by entering the option EDIT ENV from the menu following the creation of the message (see p.3). EDIT ENV will produce the TO: prompt for you to fill in with additional names.

If you wish to delete a name to which you have already addressed the message, enter a hyphen (minus sign), a space, and the name to be deleted. Note: if that name is a Folder, List, Conference, or Bulletin Board, you must enter the F, L, C, or B after the minus sign and before the name; if the person is receiving a CC or BCC, that tag must be entered following the minus sign and before the name:

```
Your command: edit env

To: - hammond
Tim S. Hammond removed from address list.
To: - cc andrews
Richard Andrews removed from address list.
To: - l budgetcom
List: Budgetcom removed from address list.
To: - c party
Conference: Party Ideas removed from address list.
```

If you wish to add or change tags (see p.4) which you have entered for a given name, you must delete the name from the envelope as described above, then re-enter the name along with the tags desired:

```
Your command: edit env

To: - cc andrews
Richard Andrews removed from address list.
To: andrews (rec
Message addressed to: Richard Andrews
To: (RETURN)

Enter a new subject line or ‹RETURN›:
Subject? (RETURN)

You may enter:
    READ   EDIT   SEND   CANCEL   FILE   MENU
```

How To: Forward A Message

After you READ a message, you will receive the following menu:

```
You may enter:
  READ     to read this message again
  ANS      to answer the sender of this message
  DELETE   to delete the message
  FILE     message in a folder
  FORWARD  to forward a message to other users
  EDIT     to edit a copy of the message
  NEXT     read next message
  MENU     return to the main menu

Your command:
```

To forward the message to another person or to a Folder, List, Conference, or Bulletin Board, choose the FORWARD option (abbreviation FO) and address the message accordingly:

```
Your command: forward
Enter message address or ‹RETURN›

To: wright
Message addressed to: Jan Wright
To: (RETURN)
```

You will be prompted for a new subject line; if you do not wish to change the original subject line, simply enter a RETURN:

```
Subject (up to 40 char): FYI; received this from Richard


You may enter:
  READ   EDIT   SEND   CANCEL   FILE   MENU
```

You may now EDIT the text of the message (see p.9), or simply SEND it to the new addressee. The message you send will show your addressing and subject line, followed by a comment line reading "Forwarded Message," and then the complete forwarded message.

How To: Search For Specific Messages

A specific message or group of messages may be located using the SCAN and READ commands. After entering the command, enter the name of the Folder, Conference, or Bulletin Board in which the message is stored, then a space, a left parenthesis, and one of the options listed below. DIALMAIL will search the envelopes of your messages, and SCAN or READ those that fit the criteria you have indicated.

The four options for searching are:

Option	Prefix	Example (from Main Menu)
Author	AUTH=	s c planning (auth=mckegney
Date	DATE=	s c planning (date=jul29
(on or before)	DATE<=	s c planning (date<=29 jul 85
(on or after)	DATE>=	s c planning (date>=7/29/85
Subject Line	SUBJ=	s c planning (subj=budget)
Message Number	MSG=	s c planning (msg=3032)

Note that a date may be entered in a variety of formats, as shown above. If no year is specified, the current year is assumed. The example below demonstrates SCANning the Conference PLANNING from the Main Menu for messages whose author is MCKEGNEY.

```
You may enter:
Scan  Read  Create  Profile  Help  Menu  Exit

Your command: scan conf planning (auth=mckegney
5 messages found in PLANNING

No.   Posted     Lines  From              Subject              Type
  1 AUG20 19:00     9 Ellen McKegney    Budget draft          Msg
  2 AUG08 19:19     8 Ellen McKegney    Second Qtr Report     Msg
  3 JUL29 21:16    14 Ellen McKegney    Statistics            Msg
  4 JUL18 11:28    14 Ellen McKegney    Proposal for budget   Msg
  5 JUN30 17:03    12 Ellen McKegney    Long-range planning   Msg
```

You may specify these search parameters any time you use SCAN or READ, whether from the Main Menu or from a Subsequent Menu. For instance, if you have already SCANned your list of Conferences to locate the Conference of interest, you may select SCAN from the subsequent menu, followed by the name of the Conference, and the search option(s) chosen, e.g.,

SCAN PLANNING (AUTH=MCKEGNEY DATE=AUG08

HELPFUL HINT: You may also use the options listed above to search your Inbox or your Desk. Just enter SCAN or READ, followed by INBOX or DESK, a space, the left parenthesis, and the prefix and data desired, e.g., SCAN INBOX (AUTH=HAMMOND.

How To: Print Mail Offline

When you SCAN your mail, you are given the option of asking that the messages be printed offline at DIALOG headquarters in Palo Alto, California, and mailed to you. The procedure appears as follows:

```
Your command: scan inbox
1 message in Inbox

No.     Posted     Lines  From                Subject              Type
  1 SEP22 17:16        7 Andy Phillips       Meeting agenda        Msg

You may enter:
    Read  Delete  Print  Scan  Help  Menu
Your command: p

Enter the numbers you would like or all: 1

Message 2041 queued for printing.
```

You may also request that a letter you create be PRINTed and sent via U.S. mail to another person. This is done by adding the word USMAIL (or USM) following a left parenthesis after the person's name at the TO: prompt (see p.4). You will be prompted for the mailing address. Be sure to enter a complete address which will be acceptable to the U.S. Postal Service.

```
To: Richard Andrews (usmail
Enter Address (up to four 50 char. lines):
? 520 Park Street, Suite 6
? Goldfield, CA 95790
? (RETURN)

Message addressed to: RICHARD ANDREWS (USMail)
To: (RETURN)
Subject (up to 40 char):
```

The message you create will be printed on letter-quality paper showing the DIALMAIL logo, inserted into a letter-size envelope, and mailed to the recipient within 24 hours from DIALOG headquarters in Palo Alto, California.

How To: Join A Conference Or Bulletin Board

You may join any Public Conference at any time. To see what conferences are available to you, use the SCAN option on the Main Menu, followed by the PUBLIC, CONFerence, and ALL options on subsequent menus. Alternatively, enter SCAN PUBLIC CONF ALL (or S P C A) from the Main Menu to skip the intermediate menus:

```
DIALMAIL - 22SEP85 17:52 EST

 You may enter:
    Scan  Read  Create  Profile  Help  Menu  Exit
 Your command: scan public conf all

 Date Begun  Organizer          Spon Public conference title

 11/06/84  Paula Steiner          Y Social committee
 11/30/84  Norm Sullivan          N Program suggestions
 09/06/84  Milt Smith             Y Restaurant suggestions
 10/23/84  Barbara Sievers        N Planning committee

 You may enter:
     JOIN to join a public conference
     SCAN to scan a public conference
     READ to read a public conference
     MENU to return to the main menu
```

To join a conference, simply choose the JOIN option. To skip a prompt for the name of the conference you wish to join, enter the first few letters of the name immediately following a space after the J for JOIN:

```
Your command: j social

You are now a member of conference: Social committee
```

HELPFUL HINT: Once you join a Conference or Bulletin Board, it is listed among all the other Conferences or Bulletin Boards to which you belong. SCAN CONF (or S C) from the Main Menu displays your list of Conferences; SCAN BBOARD (or S B) displays your Bulletin Boards. Both lists include a column showing how many NEW (unread) messages are waiting for you.

How To: Create a Conference Or Bulletin Board

To create a new Conference or Bulletin Board, use the CREATE option on the Main Menu followed by the word CONFERENCE (C) or BBOARD (B). DIALMAIL will prompt you for the remaining information:

```
You may enter:
   Scan  Read  Create  Profile  Help  Menu  Exit
Your command: create conf
Enter a short title (up to 40 char) for the conference.

Title: party ideas

Is the conference to be public(Y,N)? n

Message storage charges can either be billed to the authors or to you.

Do you wish to be billed for all message storage charges(Y,N)? n
Please enter the names of the other conference members:

Enter name: phillips
Andrew J. Phillips is now a member of the conference.

Enter name: mahoney
Sue Mahoney is now a member of the conference.

Enter name: cook
Nathan Cook is now a member of the conference.

Enter name: (RETURN)
```

If a Conference or Bulleting Board is made PUBLIC, any DIALMAIL user may SCAN, READ, or JOIN it. If not public, the originator controls which other DIALMAIL users may have access to it through the mailing list created at the end of the process shown above.

As indicated, storage for the messages in a Conference or Bulletin Board may all be charged to the originator, or may be charged to the author of each message. See ALL ABOUT CONFERENCES, p.17.

HELPFUL HINT: Enter SCAN CONF (or S C) or SCAN BBOARD (or S B) to view the list of conferences or bulletin boards in which you are currently participating!

How To: End A Conference Or Bulletin Board

You may delete any Conference or Bulletin board which you created, and withdraw from any Conference or Bboard to which you belong. For either procedure, use the SCAN option from the Main Menu followed by the CONFERENCE (C) or BBOARD (B) option. By entering S C (or S B) from the Main Menu, you will receive a response similar to the following:

```
Your command: scan conf
You have 2
Conferences:
  Budget Comments and Suggestions                19 items   3 New
  Picnic committee                                9 items   0 New

You may enter:
  SCAN          to scan messages in a conference
  READ          to read messages in a conference
  JOIN          to join a conference
  WITHDRAW      to withdraw from a conference
  DISPLAY       to display information and manage a conference
  MENU          return to the main menu
```

From the menu above you may WITHDRAW your participation from any of the conferences (or bboards) listed. To delete a conference (or bboard) that you created, use the DISPLAY option:

```
Your command  display

Enter Conference Title: Picnic

CONFERENCE
Date Created: 10/17/85 14:23:08
Organizer: Ellen McKegney
Number of Conference Messages: 9
Sponsored: Y
Public: N
Title: Picnic committee

You may enter:
    LIST        to list members
    ADD         to add a new member
    REMOVE      to remove members
    COPY        to copy the member list to another list
    END         to discontinue the conference
    PREVIOUS    return to previous menu
    MENU        return to the main menu

Your command: end
Are you sure you wish to delete conference: Picnic committee
Please enter (Y)es or (N)o: Y
```

How To: Delete A Folder Or Mailing List

To delete a Folder from your file, use the SCAN option from the Main Menu followed by FOLDER (or simply S F). The resulting menu will offer you a DELETE option, after which you will be prompted for a specific Folder name.

```
Your command: scan folder

You have 4
File folders:
Correspondence                                  6 items
Drafts                                          0 items
Budget committee communications                31 items
Marketing plan ideas                            4 items

You may enter:
SCAN        to scan messages in a file folder
READ        to read messages in a file folder
DELETE      to delete a file folder
MENU        return to the main menu
```

To delete a Mailing List which you created, enter SCAN LIST (or S L) from the Main Menu, and DELETE from the resulting menu:

```
Your command: scan list

Mailing lists:
  Budget committee
  Jogging

You may enter:
    READ        to read names in a mailing list
    DELETE      to delete a mailing list
    ADD         add names to an existing list
    WITHDRAW    to withdraw names from a list
    SEND        send a list to another user
    COPY        copy mailing list to another list
    MENU        return to the main menu
```

HELPFUL HINT: Enter DELETE (or D) followed by the name of the Folder or List, and avoid the prompt for the name! You need only enter the first few characters of the name...enough to identify it. For example, enter D JOG to delete the last List above.

ALL ABOUT: MANAGING YOUR MAIL

DIALMAIL provides you with a personal electronic desk. An INBOX receives your incoming personal mail. A DESKtop can be used to work on projects; messages left on the DESK are deleted every night. A file of FOLDERs is used for storage; messages are put in and taken out of the Folders. When you join a Conference, you gain access to the Conference folder into which all new messages are put, tagged "New" until you read them, after which they are refiled in the folder.

Messages will remain in your Inbox up to thirty days before being deleted from the system.

ALL ABOUT: FOLDERS

You are automatically given two folders in your file: CORRESPONDENCE and DRAFTS. These may be used with the FILE option to store messages received and sent, or messages which are in process.

You may CREATE additional folders through the option CREATE FOLDER (or C F) from the Main Menu. You will be prompted for a name for the folder; you may bypass this prompt by indicating the name at the time you enter the C F option (e.g., C F REPORT DRAFT).

You may also create a folder any time you use the FILE option, simply by indicating the (new) folder's name after FILE:

```
Message - Read  Answer  Delete  File  Edit  Next  Menu
Your command: file Trip report

The file folder title could not be recognized.
Do you wish to begin a new file folder with that title(Y,N)? y
Message 2426 copied to: TRIP REPORT
Message 2426 deleted from: INBOX
```

ALL ABOUT: CONFERENCES

A Conference is an ongoing folder of messages to which more than one person has access both to contribute and to read. Most Conferences have specific purposes or themes, usually reflected in their titles. The option SCAN PUBLIC CONF (S P C) from the Main Menu will produce a list of all publicly available conferences (see p.9).

Upon creating a Conference, the host user determines whether it is public or closed, and sponsored or unsponsored. A public Conference is open to any DIALMAIL user who wishes to join. A closed Conference is only available to the participants indicated by the host. If a Conference is sponsored, the host pays all message storage fees. If unsponsored, each user contributing a message pays for the storage of that message.

A Conference is a many-to-many means of communication.

ALL ABOUT: BULLETIN BOARDS

A Bulletin Board (or Bboard) is established and managed by one person. Only the host may enter new messages into a Bboard (other members may mail messages to the host for addition). If the Bboard is public, any DIALMAIL user may view its contents. If it is not public, only those users designated by the host may view the contents. The option SCAN PUBLIC BBOARD (S P B) from the Main Menu will produce a list of all publicly available Bulleting Boards. If sponsored, the host pays all message storage fees; if unsponsored, each contributor pays for the storage of his message.

The command DISPLAY from the SCAN BBOARD menu will show who is the host of a particular Bboard, and whether it is open or closed, sponsored or unsponsored.

A Bulletin Board is a one-to-many means of communication.

ALL ABOUT: DIRECTORIES

Public Directory

All DIALMAIL users are listed in the Public Directory unless they choose not to be. The Public Directory is maintained by DIALMAIL and is accessible to all DIALMAIL users. The Public Directory may not be listed in its entirety but rather may be searched for a given first or last name, or city, through the SCAN PUBLIC DIRECTORY sequence of options from the Main Menu.

HELPFUL HINT: Enter S P D xxx from the Main Menu (where xxx represents at least the first three letters of the name you are searching for) to skip the intermediate menus, e.g., S P D STEIN to see all users listed in the Public Directory whose names begin with "Stein," or S P D CHICAGO to see all users listed in Chicago.

Personal Directory

Your Personal Directory is your own address book containing the names of DIALMAIL users with whom you have corresponded. Every time you enter a name after the TO: prompt, DIALMAIL first checks your Personal Directory for the name. If it does not find the name, it checks the Public Directory, and shows you what it found there, e.g.,

```
To: embert
There is 1 person with that name in the mail system.

  Name                    Mail No    City
Valerie Embert               1483 St Louis

Is this the correct person(Y,N)? Y
Message addressed to: Valerie Embert
To:
```

Anyone who sends you a message is also automatically entered into your personal directory. To view your personal directory, enter SCAN DIR (or S DIR) from the Main Menu.

ALL ABOUT: YOUR PROFILE

The PROFILE option on the Main Menu allows you to change your user profile at any time. The use of PROFILE (or P) from the Main Menu results in the following message:

```
Mail number: 72
L)ast name: Phillips
F)irst name: Andy
O)rganization: Jamestown Information Center
A)ddress: 4051 Broadway
C)ity: Jamestown
R)egion: State or Country: ME
Z)ip or Mail Code: 99999
P)ublic directory listing(Y,N): Y
S)ize of menu (Long,Short,None): L
D)isplay number of lines on screen: 40
N)umber of days messages left in Inbox (1-30): 30
I)nbox message disposition (Delete, Print, File, Forward): D
V)iew profile
M)ain menu

Your command:
```

By entering the first letter of any of the options, you may change it. For instance, to permanently change the size of menus from LONG to SHORT, follow the sequence below after you receive the prompt above:

```
Your command: s

Menu length: (Long,Short,None)? short

Your command:
```

You might wish to use the VIEW option to review your Profile following any changes you make.

ALL ABOUT: FORMS and ILL

When you enter the CREATE option from the Main Menu, two of the options available are FORMS and ILL. FORMS is a capability which allows a form to be created (with empty fields for future input) and stored on DIALMAIL for later use, while ILL is a form permanently set up within DIALMAIL for use in interlibrary loan applications. In short, FORMS allows you to create your own forms, while ILL provides a form for you to use in a specific situation.

You can create a form by entering CREATE FORM from the Main Menu. You will first receive the TO: prompt. You may send your form to other people, to a Conference or Bulletin Board, or to a Folder. You will then be prompted for a subject line, and then for the text of the form. Simply enter the text the way you want it to appear. At any location where you want to leave space for future response, enter a colon (:), followed by the number of spaces plus one needed for the response. If more than one line is needed for response, enter a colon by itself at the beginning of each line of space desired followed by a (RETURN).

Once you have finished entering the text of the form, enter a period (.) on a blank line. You may then EDIT or SEND the form to its destination.

An example of this process is shown below:

```
Your command: create form
Creating Form:

To: f mileage form
Message addressed to file folder: MILEAGE FORM
To: (RETURN)

Subject: (up to 40 char): Weekly Mileage Report
Enter text, ending it by typing a period on a blank line.

                  REPORT FORM FOR MILEAGE
Week ending:
Employee name:
Department:
MILEAGE
  Personal car
      From:
        To:
     Miles:
   Purpose:

  Company car:
      From:
        To:
     Miles:
   Purpose:

Comments:
.
```

When you receive a form in your Inbox, READ it first. From the menu following READ, enter the ANSWER option. After addressing your response to its recipient, you will receive the elements of the form one by one, with a pause at each colon for you to fill in the information. At the end of the form, DIALMAIL will prompt you to either EDIT or SEND your response. The ILL form works in this same way, except that you enter CREATE ILL from the Main Menu rather than READ the form from your Inbox.

ALL ABOUT: MAIN MENU COMMAND OPTIONS

Main Menu command options may be entered followed by options from subsequent menus in order to skip the latter menus. The Main Menu option is entered followed by a space and the subsequent menu option(s). Options may be abbreviated to their first character. Here is a complete list of the command sequences that may be issued from the Main Menu, with a description of the results of each:

SCAN	
INBOX	Lists messages in your Inbox
DESK	Lists messages on your Desk
FOLDER	Lists all your Folders
FOLDER xxx	Lists messages in Folder xxx
DIRECT	Lists your Personal Directory
DIRECT xxx	Lists entry xxx from your Personal Directory
CONF	Lists Conferences to which you belong
CONF xxx	Lists contents of Conference xxx
BBOARD	Lists Bboards to which you belong
BBOARD xxx	Lists contents of Bboard xxx
PUBLIC	Shows Public menu options
PUBLIC CONF	Lists Public Conferences
PUBLIC CONF xxx	Lists contents of Public Conference xxx
PUBLIC BBOARD	Lists Public Bulletin boards
PUBLIC BBOARD xxx	Lists contents of Public Bboard xxx
PUBLIC DIR	Prompts for name to scan for in Public Directory
PUBLIC DIR xxx	Lists entries xxx in Public Directory
READ	
INBOX	To read messages in your Inbox
DESK	To read messages on your Desk
FOLDER xxx	To read messages in Folder xxx
CONF xxx	To read messages in Conference xxx
BBOARD xxx	To read messages in Bboard xxx
CREATE	
MESSAGE	Creates a message for mailing
FOLDER	Creates a personal file Folder
FOLDER xxx	Creates Folder; bypasses title prompt
CONF	Creates a Conference
CONF xxx	Creates Conf; bypasses title prompt
BBOARD	Creates a Bulletin board
BBOARD xxx	Creates Bboard; bypasses title prompt
ALIAS	Creates an alias for a name
LIST	Creates a mailing list
FORM	Creates a form
ILL	Creates an interlibrary loan form to fill out and send
HELP	
HELP	Shows all help options
HELP xxx	Gives HELP message for command xxx

CONNECTING TO DIALMAIL

For DIALOG Accounts

Users of the DIALOG or DIALOG Version 2 system may use their regular DIALOG passwords for access to DIALMAIL. To connect to DIALMAIL, enter the appropriate service address for the network you are using (see below). When you receive the prompt PLEASE LOGON: simply enter your password, and you will be connected to DIALMAIL. Note that the logon to DIALMAIL is separate from the logon to either DIALOG or DIALOG Version 2.

Service Addresses for DIALMAIL

Within the United States:

VIA DIALNET: DIALMAIL

VIA TYMNET: DIALMAIL

VIA UNINET: DIALOG;MAIL or DLG;MAIL or DLG;M

VIA TELENET: C 41520M or C 41548M or C 213170M or C 213236M

Outside the United States:

The exact method of connection and the exact network address for DIALMAIL will depend upon your location and the protocols of your telecommunication network. The addresses below are the standard international addresses for the networks indicated, and must be adapted for your particular locale.

VIA DIALNET: DIALMAIL

VIA TYMNET: 3106900803,I

VIA UNINET: 31254150005000,M

VIA TELENET: 31104150004813

For KNOWLEDGE INDEX Accounts

Users of KNOWLEDGE INDEX have access to DIALMAIL from within the KI service. After following regular logon procedures, simply enter BEGIN MAIL to connect to DIALMAIL.

DIALMAIL is not available betweeen the hours of 05:00 and 07:00 Eastern Standard Time, Monday through Saturday, nor on Sunday from midnight (00:00) to noon (12:00) EST.

AFIPS Conf. Proc, vol. 45 pp 109-112

Multiuser cryptographic techniques*

by WHITFIELD DIFFIE and MARTIN E. HELLMAN
Stanford University
Stanford, California

ABSTRACT

This paper deals with new problems which arise in the application of cryptography to computer communication systems with large numbers of users. Foremost among these is the key distribution problem. We suggest two techniques for dealing with this problem. The first employs current technology and requires subversion of several separate key distribution nodes to compromise the system's security. Its disadvantage is a high overhead for single message connections. The second technique is still in the conceptual phase, but promises to eliminate completely the need for a secure key distribution channel, by making the sender's keying information public. It is also shown how such a public key cryptosystem would allow the development of an authentication system which generates an unforgeable, message dependent digital signature.

INTRODUCTION

In a computer network with a large number of users, cryptography is often essential for protecting stored or transmitted data. While this application closely resembles the age old use of cryptography to protect military and diplomatic communications, there are several important differences which require new protocols and new types of cryptosystems. This paper addresses the multiuser aspect of computer networks and presents ways to preserve privacy of communication despite the large number of user connections which are possible.

In a system with n users there are n^2-n pairs who may wish to hold private conversations. The straightforward way to achieve this is to give each pair of users a key in common which they share with no one else. Each user will then have n-1 keys, one for communicating with each other user. Unfortunately, the cost of distributing these keys is prohibitive. A new user must send keys to all other users. Unfortunately, the network cannot be used for this purpose, and an external secure channel is required. This procedure is comparable to requiring each new telephone subscriber to send a registered letter to everyone else in the phonebook.

Military communications suffer less from this problem for several reasons. Among these are the limitations imposed by the chain of command and the fact that stations change allegiance infrequently. In a computer network designed for business communication, on the other hand, users will regard each other as friends on one matter and as opponents on another. Firms A and B may cooperate on one venture in competition with C, while simultaneously, A and C compete with B on a different endeavor. A must therefore use different keys for communicating with B and C.

One approach to this problem is to assume that the users trust the network. Each user remembers only one key which is used to communicate with a local node. From there the message is relayed from node to node, each of which decrypts it, then reencrypts it in a different key for the next leg of its journey. This process is known as link encryption.[1] When the message reaches the network node closest to its destination, it is sent on to the addressee encrypted in a key shared only by the addressee and that node.

Although this technique requires each user to remember only one key, it has the disadvantage that a message is compromised if any one of the nodes in its path is subverted. In this paper we examine two other ways of allowing secure communication between any pair of users without assuming the integrity of all nodes in the network and without requiring the users to distribute or store large numbers of keys.

The first technique requires no new technology, but imposes a complex initial connection protocol. This is the subject of the second section of this paper. We call the second technique public key cryptography since most of the secrecy traditionally required for the keys has been removed. This is discussed in section three and represents a radical departure from past cryptographic practices. While it requires further work before it becomes implementable, its simplicity of operation makes it extremely attractive. If a suc

* This work was supported by the National Science Foundation under NSF Grant ENG 10173.

cessful implementation can be developed it should find wide use in both military and civilian applications.

The fourth section shows how public key cryptography can be used to provide a time and message dependent digital signature which cannot be forged even when past signatures have been seen. This is an example of the general problem of authentication discussed in greater detail in Reference 2, which provides a more general perspective in which public key cryptography can be viewed.

A PROTECTIVE PROTOCOL

As indicated earlier, a message protected by link encryption will be compromised if any node in the path it follows from the sender to the receiver is subverted. In this section we describe a protocol which guarantees to protect the message unless a large number of nodes are compromised. While many variations are possible, the basic technique is as follows.

A small number m of the network's nodes will function as "key distribution nodes." Each user has m keys, one for communicating with each of these m nodes. These keys vary from user to user, so while each user must remember only m keys, each of the key distribution nodes remembers n, one for each user of the net. When users A and B wish to establish a secure connection they contact the m key distribution nodes and receive one randomly chosen key from each. These keys are sent in encrypted form using the keys which the users share with the respective nodes. Upon receiving these keys, the conversants each compute the exclusive or of the m keys received to obtain a single key which is then used to secure a private conversation. None of the nodes involved can violate this privacy individually. Only if all m nodes are compromised will the security of this connection fail.

It might be objected that any key distribution node acting alone can prevent all communication by mischievously sending out different keys to each of the parties, thus bringing network operations to a halt. The users, however, can easily protect themselves against this threat. If communication using the composite key fails, its use as a key is abandoned, and the components are exchanged one by one, in clear, for comparison. If any key fails to agree, the node which issued it is blacklisted. Finally, on conclusion of this process, the users repeat the request for keys to the nodes which passed the previous test.

Alternatively, the component keys can be compared by the use of one way functions[2,3,5] without ever being transmitted in clear. Loosely speaking, a function f is called a one-way function if it is easy to compute in the forward direction, but given any output, it is computationally infeasible to find an input which produces it. In referring to a task as computationally infeasible, we have in mind that it cannot be done in fewer than a finite but astronomical number of operations, say $2\uparrow 100$. For practical purposes, this is equivalent to being incomputable. As shown in Reference 2, a one way function can easily be obtained from a secure cryptosystem.

If communication fails using the composite key, the users send the images of the individual keys under a public one-way function. If the image received does not agree with that computed by applying f to the key, the node which issued it is guilty of compromise. Since the valid keys have not been publicly revealed in this process, there is no need to request new ones from the uncompromised nodes. Instead the invalid ones are omitted and the remainder xored.

To sum up, this technique requires each user to remember m keys and each key distribution node to remember n keys. Unless all m key distribution nodes are subverted, any two users can establish a private link through use of a set-up protocol usually requiring 2m exchanges (more are required if a key distribution node has been subverted). The next section describes a concept which eliminates much of this overhead and does not require the user to trust any node. This new concept, if successfully implemented, will make the technique described above obsolete.

PUBLIC KEY CRYPTOGRAPHY

In this section we propose that it is possible to eliminate most of the secrecy surrounding the key used in a communication, and yet to preserve the secrecy of the communication. This is accomplished by giving each user a pair of keys E and D. E is an enciphering key and is public information. D is the corresponding deciphering key, and while this must be kept secret, it need never be communicated, eliminating the need for a secure key distribution channel. Although D is determined by E, it is infeasible to compute D from E.

For reasons of security, generation of this E-D pair is best done at the user's terminal which is assumed to have some computational power. The user then keeps the deciphering key D secret but makes the enciphering key E public by placing it in a central file along with his name and address. Anyone can then encrypt a message and send it to the user, but only the intended receiver can decipher it. Public key cryptosystems can therefore be regarded as multiple access ciphers.

By regularly checking the file of enciphering keys the user can guard against any attempt to alter it surreptitiously. Any such mischief is reported and settled by other authentication means, such as personal appearance.

The crucial feature of a public key system is that it is relatively easy to generate an E-D pair, preferably automatically through a publicly available transformation from a random bit string to E-D, and yet it is computationally infeasible to compute D from E.

At present we have neither a proof that public key systems exist, nor a demonstration system. We hope to have a demonstration E-D pair in the near future, and expect that if the demonstration pair successfully resists attack then we will be able to design an algorithm for automatically generating E-D pairs of a similar kind. In the meantime, the following reasoning is given to help dispel any doubts the reader may have.

A suggestive example is to let the cryptogram, represented as a binary n-vector c equal E m; where m is the message also represented as a binary n-vector, and E is an arbitrary n-by-n invertible matrix. Letting D=E↑-1 we have m=D c. Thus both enciphering and deciphering are easily accomplished with about n↑2 operations. Calculation of D from E, however, involves a matrix inversion which is a harder problem. And it is at least conceptually simpler to obtain an arbitrary pair of inverse matrices than it is to invert a given matrix. Start with the identity matrix I and do elementary row and column operations to obtain an arbitrary invertible matrix E. Then starting with I do the inverses of these same elementary operations in reverse order, to obtain D=E↑-1. The sequence of elementary operations could easily be generated from a random bit string.

Unfortunately, matrix inversion takes only about n↑3 operations even without knowledge of the sequence of elementary operations. The ratio of "cryptanalytic" time (i.e., computing D from E) to enciphering or deciphering time is thus at most n. To obtain ratios of 10↑6 or greater would thus require enormous block sizes. Also, it does not appear that knowledge of the elementary operations used to obtain E from I greatly reduces the time for computing D. And, since there is no round-off error in binary arithmetic numerical stability is of no consequence in the matrix inversion. In spite of its lack of practical utility, this matrix oriented example is still useful for clarifying the relationships necessary in a public key system.

A more practical direction uses the observation that we are really seeking a pair of easily computed inverse algorithms E and D, but that D must be hard to infer from E. This is not as impossible as it may sound. Anyone who has tried to determine what operation is accomplished by someone else's machine language program knows that E itself (i.e., what E does) can be hard to infer from E (i.e., a listing of E). If the program were to be made purposefully confusing through addition of unneeded variables, statements and outputs, then determining an inverse algorithm could be made very difficult indeed. Of course, E must be complicated enough to prevent its identification from input-output pairs.

Another idea appears more promising. Suppose we start with a schematic of a 100 bit input, 100 bit output circuit which merely is a set of 100 wires implementing the identity mapping. Select 4 points in the circuit at random, break these wires, and insert AND, OR and NOT gates which implement a randomly chosen 4 bit to 4 bit invertible mapping (a 4 bit S box in Feistel's notation).[1] Then repeat this insertion operation approximately 100 times to obtain an enciphering circuit E. Knowing the sequence of operations which led to the final E circuit allows one to easily design an inverse circuit D. If however the gates are now randomly moved around on the schematic of E to hide their associations into S boxes, an opponent would have great difficulty in reconstructing the simple description of E in terms of S boxes, and therefore would have great difficulty in constructing a simple version of D. His task could be further complicated by using reduction techniques (e.g. Carnaugh maps) or expansion techniques (e.g. ~(AB)=~A or ~B, or expressing a logical variable in terms of previous variables), and by adding additional, unneeded S boxes and outputs.

For ease of exposition, we have described the implementation of a specific key in hardware. In practice, a special purpose simulator is obviously of most interest. The hardware description is also valuable in exemplifying a generally useful idea. To build a good public key cryptosystem one needs easily inverted elementary building blocks and a general framework for describing the concatenation of these elementary blocks. Here the elementary building blocks are S boxes and the general framework is the schematic diagram. The general framework must also hide the sequence of elementary building blocks so that no one other than the designer can easily implement the sequence of inverse elementary operations. Examination will show that the matrix example had a similar structure, except there the general class of transformations obtainable was too small.

While the above arguments only provide plausibility as opposed to proof, we hope they will stimulate additional work on this promising area of research.

PUBLIC KEY AUTHENTICATION

The purpose of a cryptographic system is to prevent the unauthorized extraction of information from a public (i.e., insecure) channel. The dual problem of authentication is to prevent unauthorized injection of messages into a public channel.

In conventional paper oriented business transactions, signatures provide a generally accepted level of authentication. As electronic communication replaces mail service the need for a digital signature will be strongly felt.

Various types of authentication are now possible,[2] but the development of public key cryptosystems would allow an entirely new dimension.

Currently, most message authentication consists of appending an authenticator pattern, known only to the transmitter and intended receiver, to each message

and encrypting the combination. This protects against an eavesdropper being able to forge new, properly authenticated messages unless he has also stolen the key being used. There is no protection against such an eavesdropping thief or against the threat of dispute. That is, the transmitter may transmit a properly authenticated message, later deny this action, and falsely blame the receiver for taking unauthorized action. Or, conversely, the receiver may take unauthorized action, forge a message to itself and then falsely blame the transmitter for these actions. For example, a dishonest stockbroker may try to cover up unauthorized buying and selling for personal gain by forging orders from clients. Or a client may disclaim an order, actually authorized by him, but which is later seen to cause a loss. We will introduce concepts which would allow the receiver to easily verify the authenticity of a message, but which prevent him from generating apparently authenticated messages, thereby protecting against both the threat of eavesdropping thieves and the threat of dispute. Note that these techniques thus provide stronger protection than signatures, voiceprints, etc. which can be forged once seen and are not message dependent.

To obtain an unforgeable digital signature from a public key cryptosystem, the protocol would be as follows: Assume user A wishes to send a message M to user B. The transformed message C=EbDa(M) is sent, where Eb represents the transformation effected by use of B's public enciphering key and Da represents the transformation effected by use of A's secret deciphering key. Upon receipt of C, user B operates first with his secret operation Db and then with the public operation Ea thereby obtaining EaDb(C) = EaDbEbDa (M) = M. No one else can extract M because of the need to know Db. By saving the intermediate result Db(C) = Da(M) user B (and only user B) can prove that he received the specific message M from user A. There must be some structure to the message (e.g., it could include a date and time field) to prevent injection of random bit patterns for C, with the hope that the resultant decoded "message", EaDb(C), might cause random mischief such as deletion of files.

Note that since there is no need for a secure channel for distribution of authentication information, we have a public key authentication system. This system protects against, "eavesdropping thieves" and against a dispute as to whether or not an action taken by the receiver was authorized by the transmitter. Similarly, a public key cryptosystem can be used to protect against the other type of dispute in which the transmitter A claims to have issued an order which was not carried out by the receiver B. The transmitter requests that the receiver B send EaDb(M) as a receipt for the message M. By operating on this receipt with his secret operation Da, the transmitter obtains Db (M), which could only have been generated by the receiver B. Only user A can generate this receipt since it requires knowledge of Da.

While the above discussion centered on message authentication it also applies to user authentication. The implicit message becomes "I am user X and the time is T." Inclusion of the time field prevents an eavesdropper from using old authentication signals to pose as someone else. For reasons noted in Reference 2, such a system deserves to be called a one-way IFF system.

We thus see that public key cryptosystems developed for ensuring the privacy of communications, could also be used to ensure their authenticity. They could therefore be used to fill the need for a digital equivalent of a signature. This need is currently a major barrier to the use of electronic mail for business communications, and provides additional motivation for study of public key cryptosystems.

ACKNOWLEDGMENT

The authors wish to thank Leslie Lamport of Massachusetts Computer Associates for several valuable discussions. In particular, the technique described in Section 2 was discovered during one of these conversations.

REFERENCES

1. Baran, Paul, *On Distributed Communications: IX. Security, Secrecy, and Tamper-Free Considerations*, Santa Monica, CA: The Rand Corporation August 1964, (RM-3765-PR).
2. Diffie, Whitfield and Martin E. Hellman, forthcoming paper to be submitted to the *IEEE Transactions on Information Theory*.
3. Evans, Arthur, Jr., William Kantrowitz and Edwin Weiss. "A User Authentication System not Requiring Secrecy in the Computer," *Communications of the ACM*, Vol. 17 No. 8, August 1974, pp. 437-442.
4. Feistel, Horst, "Cryptography and Computer Privacy," *Scientific American*, Vol. 228, No. 5, May 1973, pp. 15-23.
5. Purdy, George B., "A High Security Log-in Procedure." *Communications of the ACM*, Vol. 17, No. 8, August 1974, pp. 442-445.

DIGICASTING FOR THE HEARING-IMPAIRED

Dear Jim, 78 Nov 25

I met you briefly last March at the San Jose Computer Faire (using my wheelchair, you may recall). After teaching junior high age deaf students since 1962, I am presently on a half-year study leave (Sabbatical) to investigate "Computers/Telecommunications in Special Education."

As part of this study, I recently toured the east for five weeks which included a visit to the RTTY News Service for the deaf at the Pennsylvania School for the Deaf in Philadelphia, where they prepare daily newscasts, broadcast them via FM sub-carrier to a 50-mile-radius audience of deaf persons who receive the news via a small FM receiver/sub-carrier decoder which is then printed out on the home TTY (5-level Baudot) which is also used with a modem for two-way conversations via telephone lines with other deaf persons who have similar equipment.

This is a marvelous service to the deaf community in Philadelphia.

Yet, here in the Bay Area with hundreds, if not thousands, of 5-level Baudot TTY's in homes, schools and agencies of the deaf, such a service does not exist.

Your Digicast project is a most promising one.

Would it be possible for Digicast to broadcast in 5-level Baudot for, say an hour a day, using hardware or software conversion from 8-level ASCII, in order to provide a greatly needed service for the deaf in the Bay Area Community?

Being an educator rather than a technician, I am unsure if my request is a simple or complicated one, but I sincerely ask you to consider it.

Grant Grover
Office of Marin County
Superintendent of
Schools
201 Tamal Vista Boulevard
Corte Madera, CA 94925
(415) 924-9500

DIGICAST PROTOCOLS DESIGN WORKSHOP SCHEDULED

The Digicast Project, working in cooperation with Professor Edwin Parker of Stanford University's Communication Department, has scheduled a two-day Workshop to address the design of transmission, communication, and format protocols for Digicasting transmitters, receivers, and data. Participation will be by invitation-only, and will be limited to approximately 20 individuals. It will take place at:

Stanford University
January 18-19, 1979
(Thursday-Friday).

Computer and communications professionals are encouraged to request invitations who have expertise or experience in any of the following areas:

broadcasting technology
transmitter design
receiver design
communication protocols
digital transmission
error detection & correction
information indexing & cataloging

The goal of this workshop is to produce: (1) a list of technical problem areas impacting implementation of Digicasting systems; (2) proposed solutions to those problems; (3) explicit, completely-specified protocols, where such selection and specification is clearly reasonable; and (4) lists of alternative protocols where a single best choice is not self-evident. The expectation is that this Workshop, with zero or few iterations, can initiate the complete specification of the transmission protocols, from "digital input" at the transmitter, to "digital output" at the receiver, and can at least specify the lowest-level data formatting protocols, e.g., "EOA" codes – End of Article.

Requests for invitations should include a very brief statement of related experience and expertise, and be directed to:

Jim Warren, Director
Digicast Progect
345 Swett Road
Woodside, CA 94062
(415) 851-7075

This is a production-oriented working session, not an educational or tutorial program. Once the Project is firmly under way, there will be seminars, tutorials, and conference papers, designed to provide background and details for the interested amateur, and newly-arriving professionals and entrepreneurs.

Intelligent Machine's Journal; Dec 11, 1978