XERPT

VOL. 1, NO. 2 A Publication of Xerox Printing Systems FALL 1979

Ethernet Network Delivers The Mail

Distributed data processing is a concept hampered by an identity crisis: there's a computer configuration in it for every Tom, Dick and Mary. As a cocktail party buzzword DDP ranks high, but as a descriptive term it falls short.

DDP's problem reflects a larger debate within the computer industry over the merits of centralization versus decentralization. Centralizers see the eternal demand for mainframes, while decentralizers envision a world of personal computers for all.

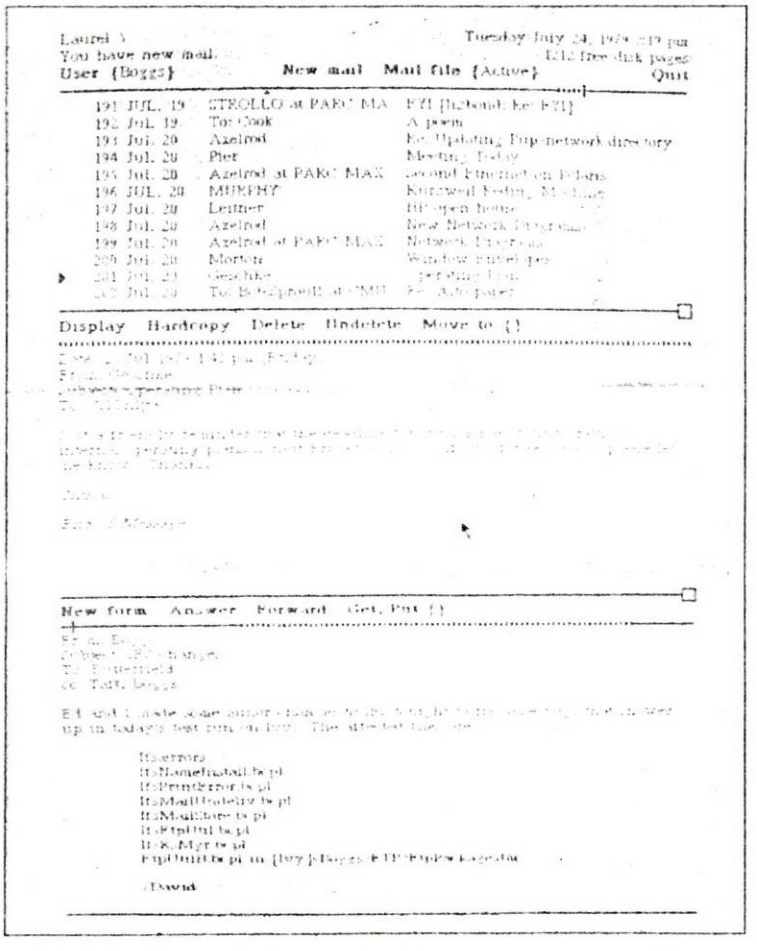
Whatever lies between tends to be labeled distributed data processing.

Actually, the DDP spectrum ranges from remote computer networking, a loose interconnection of large, widely separated computing systems, to multiprocessing, a computer configuration comprised of previously independent systems which access a common memory in parallel. A DDP middleground called local networking is based on the premise that if computer networks can cross continents and oceans they should also be able to cross corridors, interconnecting computers in offices and laboratories. In theory, the local network combines the resource sharing of computer networking with the parallelism of multiprocessing.

Which brings us to the Ethernet network, an experiment in local networking pioneered in 1972 at the Xerox Palo Alto Research Center (PARC) and since matured into an *internet (italicized words are described in a glossary on page 4)* communications system used daily by several thousand Xerox employees scattered across the country.

Technically, the Ethernet system is a branching broadcast communications network for carrying digital data packets among locally distributed stations. Its main parameters include up to 256 hosts (or stations), a data transfer rate of 3 megabits per second and a locally operational environment of 1 kilometer. Packets consist of approximately 300 computer words or 4,500 bits and resemble a letter, with an address, message and return address.

Informally, the system is simply a way of moving bits around in a campus-type environment to enable people working at personal computers to talk to each other; hence, a very reliable, inexpensive distributed computing system that grows smoothly over succeeding magnitudes of size without entailing a large jump in economics or allowing a single point failure to cause the whole network to go down.



An Ethernet network electronic mail program CRT display shows list of David R. Boggs' messages (upper panel), message from "Chuck" (center panel) and Boggs' message to "Butterfield." Photographed directly from a CRT.

That the Ethernet network lacks any centralized control is indisputable: there is no central computer. To eliminate the reliability problems of an active central controller, everyone has his or her own minicomputer — the vast majority are called *Altos* — and they're all equal. The shared communication line connecting them is a continuous piece of coaxial cable, or "Ether," which acts as a passive broadcast medium, switching data packets of varying bit lengths to any *network address* in the system. A station's Ethernet network interface connects bit-serially through an interface cable to a transceiver which in turn taps into the passing "Ether." A packet broadcast onto the "Ether" (so named for the medium once thought to have permeated the universe) is heard by all stations but copied only by the stations that like its leading address bits. Statistical arbitration controls access to the "Ether" among the stations.

Such an explanation of how this system works may well satisfy computer technicians and professionals, but too much 'controlled statistical arbitration' could drive the average user to drink.

"Fair enough," says David Boggs, one of the system's inventors and a PARC research scientist (see profile, page 2), "picture yourself at a cocktail party with six or eight people standing around in a circle talking. Think of each one as a computer hooked to the Ethernet network, the air

Continued on page 2

Continued from page 1

within the circle, in this example.

"Someone is talking while the others are halfway listening, waiting for him to finish so they can tell their story. The problem is how to mediate who gets to talk next. You don't want one guy to act as a central controller telling everyone else who gets to talk. Thus as one person finishes his story, everyone looks around and starts talking, often at the same time.

"Hearing each other's voices, they all stop, eye each other, and start talking again. Eventually, someone 'acquires the channel' and gets to tell his story. When he's finished, the process repeats."

A cocktail party also serves as a sobering metaphor for explaining what happens when two or more stations attempt to send a packet at roughly the same time.

"Each time two or three people start talking, hear each other and stop," David says, "that's a collision. Random transmission is everyone eyeing everyone else trying to decide who will talk. Carrier sense is knowing not to talk when someone else is talking. Collision detection is hearing someone else talking.

"In a sense it's a game. If each person plays by the rules, everyone will get his fair share of conversation. If one guy cheats, he can monopolize the conversation."

Actually, as more stations begin to transmit and the rate of packet interference increases, controllers in each station adjust the mean retransmission interval in proportion to the frequency of collisions, thereby optimizing sharing of the "Ether." Still, collisions do occur. There is no guarantee that a packet will arrive undamaged or, for that matter, will arrive at all.

"It's not perfect," David says. "It's a passive medium designed to achieve reliability through simplicity. You give it a packet. It does its best to deliver it. When you send a packet, you keep a copy until you know it has arrived in good shape. When we need reliability we can get it, very simply, by adding protocols to cover such contingencies as packets that don't arrive or arrive damaged. Incidentally, packet

Continued on page 3

Profile:

Boggs: 'Mainframes are dinosaurs'

The features are intellectual.

The clear-blue eyes peer intently over plain-rim glasses. Suddenly the voice is forceful, strengthened by the righteousness of a strong conviction.

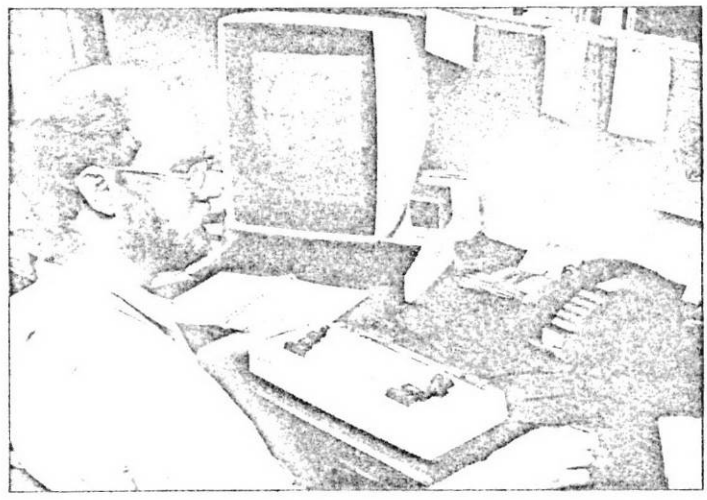
"Distributed is what we're trying to do around here," he says, the tone of his words at odds with the casual, professorial demeanor. "We invented distributed processing. I'm not saying we invented the term," he pauses, "but we should have."

David R. Boggs swivels around to face his Alto computer terminal. Lean fingers jog quickly over the keyboard. A near perfect photographic likeness, like a large, screened halftone, flashes on the CRT — a program he wrote to inform PARC colleagues 'dialing' his network address that his latest project was not yet ready to share.

"What most people mean by distributed data processing," he says, "is a central computer with a lot of terminals hanging off of it. We take it one step further: no central computer at all.

"Mainframes are dinosaurs. They're already dead only they don't know it yet. In the bad old days, when the big number cruncher went down everyone stopped working. Couldn't do anything. We wanted better: if your computer goes down, *you're* out of luck, but a hundred other people in the building can continue to work. It seems like human nature to pile everything in one place. We're trying to get it down to the very minimum, but as soon as you look the other way somebody will try to recentralize it."

At 29, degrees from Princeton and Stanford behind him, a Ph.D. within reach by year's end, David Boggs already is a recognized pioneer of local networking, entitled to, and plainly enjoying, the perquisites allowed creative and productive innovators at a heaven-sent research environment such as PARC.



PARC scientist David R. Boggs, at keyboard of his Ethernet network Alto computer, is closely observed by David R. Boggs (barely visible on CRT).

He is a 30-message-a-day man, which adds up to a lot of electronic mail, even for one of the inventors of this successful experiment in personal computing. They arrive from practically every station on the Xerox internet — queries, announcements, diatribes, arguments, pleas for explanations, for esoteric data, for personal appearances at meetings and seminars. He keeps a complete running hardcopy file of them, in special black folders, "in case, someday, a psychologist wants to study the communications of Ethernet network users." Clearly, he hopes some psychologist will.

Robustly, and with engaging frankness, David returns to his central (make that decentral) theme: "There's a lot of lip service these days about distributed this and distributed that, but do people really believe it? Look closely at DDP. It's really just jazzier terminals with another Big Mama back in the bunker somewhere. When it goes down, everything stops. At PARC, we think the failure of a piece of hardware should inconvenience as few people as possible. In fact, we get nervous when more than 12 people depend on one machine. We'd rather have an identical machine somewhere else within walking distance."

A city boy without wilderness needs, David was born and raised in Washington, D.C. At 14 he became an amateur radio operator, later picking up a commercial operators license first used at Princeton, where he was chief engineer of the University's commercial FM station, and then employed at NBC, where he was a vacation relief engineer.

In September 1972 he came to California to get an M.S. in electrical engineering at Stanford. In the process, he began hanging around PARC, pestering people, in his own words, for a job. He was hired by a PARC group working on a centralized office system but was quickly borrowed, permanently as things turned out, by then Xerox scientist Robert M. Metcalfe, who asked him to collaborate on the Ethernet communications network concept.

Finishing his Ph.D. and building a Lotus Super 7 automobile from a *kit*, David still finds time to tinker around with the stock market. Someday he wants to rent a telephone line from his Los Altos Hills home directly to the Stock Exchange so he can get his hands on all the tickertape transactions and write a program to analyze them.

If that program ever gets written, bulls beware — here comes Boggs. □

THE NOT-SO-REMOTE PRINTER SALESMAN

Sherry: 'I got my job through The NY Times'

One crisp Sunday afternoon in the fall of 1972, Bob Sherry finished tinkering with the 1931 model A in the backyard of his Norwalk, Connecticut home, settled down with five pounds of "New York Times" and found himself a career.

The advertisement jumped off the page as if it had been written for Bob alone: Xerox was looking for people knowledgeable in Scientific Data Systems (SDS) computers.

Bob knew SDS computers from circuitry to software. He had programmed them, trained programmers to use them and had helped operate a time-sharing system built around an SDS system at Perkin-Elmer Corp., where he had spent three years as a member of a small dp group after getting his B.A. from the University of New Haven.

Should he get back into computers, or should he remain in his own business, refurbishing used cars for 21 Norwalk automobile dealers? The refurb business was a nice living, especially if you enjoyed wearing dirty jeans all day six days a week.

Opting for clean clothes, Bob joined Xerox two days before Thanksgiving 1972.

Bob Sherry is a tall, friendly, sports-loving man who fails to

BOB SHERRY

contain a boyish exuberance—doesn't even try—for his work. "The technology here is unbelievable; it's mindboggling. We sometimes don't understand how good it really is—the quality, the flexibility, the reliability. Take the 9700, for example . . ."

"Bob," interrupted the XERPT reporter, "this is a profile about you, not the 9700."

"O.K. I spent my first two years with the company on a product called Biscom, a system designed to help A.T.&T. simplify the paperwork for changing and moving telephones. New York Telephone was losing hundreds of orders a day because of paperwork problems. Biscom fed all these changes into a central computer through a host of CRTs and solved the problem. I learned a lot; it was a lot of fun.

"Then I joined the Computer Printing Group as an analyst, working on the 2400 duplicator and the 1200 printer, the first non-impact computer printer. I spent most of my time programming the 1200 to read magnetic tape." Although he has little use for them today, Bob 'speaks' Fortran, COBOL, Metasymbol and BAL.

Bob first hit the street in 1977, selling the 1200 in a territory that included Fairfield County, Conn., and Westchester County, New York, an area he had dribbled through hundreds of times as a basketball guard for Norwalk State College, where he first studied computers, and for a championship Greenwich, Conn. YMCA team.

Selling came naturally. He won "Rookie of the Year" honors in '77, finishing seventh nationally in divisional sales.

Following Thanksgiving Day last year, Bob was promoted to Regional Terminal Specialist (RTS) for the Eastern Region, making him responsible for training the 30-person regional sales force, seeing to it that everyone makes his or her terminal quota, instilling a pinch of his enthusiasm for work—and life!

"We're a lean division. We need people who appreciate the excitement of our products, who are not afraid to work, who like challenges."

'Challenge' fits comfortably in Bob Sherry's vocabulary. He and wife Pat play competitive tennis, attend twenty or thirty antique car shows a year, coach a girl's softball team, jog the beaches of Connecticut and encourage daughters Jennifer, 7, who wants to be an ice skater, and 11-year-old Alicia, a budding violinist.

Not surprisingly, Bob is upbeat about the printer terminal business. "Computer printing didn't really change a whole lot from its inception until about 1973. Sure, it became a little faster, but basically it remained the same. People gradually got used to the large-size paper output and stopped thinking about it. Now Xerox comes along and says, 'We can give you flexibility in paper size, whatever size you want.'

"Our printing technology is superb. This is going to be a huge division. It's really a good place to be—a real challenge." □

Ethernet Network

Continued from page 2

errors tend to affect only one station, not the whole network."

PARC technologists have invented quite a few simple and elaborate protocols, the rules *Alto* computers use in talking to one another. Among them are protocols for transferring files, emulating terminals and sending and receiving electronic messages. A packet is a transient thing, a burst of bits rarely amounting to more than a fragment of conversation. Many of them, numbered in sequence, must be reassembled into a conversation by a receiver. The substance of a conversation is yet another protocol.

Sending and receiving individual packets to any *network address* beyond the 1 kilometer local environment is accomplished through Xerox's far-flung *internet* system called the *"Pup"* Internet, which connects some 1000 computers on 25 Ethernet systems with 20 internetwork *gateways* (computers that serve as forwarding links between separate Ethernet networks). "Pup" consists of a whole hierarchy of internetwork protocols supported by powerful personal computers, minicomputer systems and various specialized systems.

Electronic mail is in the forefront of this burgeoning technology. *"Pup"* makes electronic mail correspondence a daily reality for some 3000 Xerox employees located within the 25 separate Ethernet network locations that comprise the National Alto Network (see map, page 4).

Using a text editor, a Dallas sender, for example, composes a message. When sent over the Ethernet network, as noted earlier, the message is cut up into smaller, numbered packets and transmitted, fragment by fragment, to its Dallas mail server, which reassembles the message, inspects its destination address and decides to dispatch it to, say, Rochester, through a gateway. The Rochester *mail server* inspects the destination address and adds it to the recipient's message file. When the recipient arrives at his office, he runs a mail program on his *Alto*, which retrieves and displays his messages.

Inevitably, the question is raised whether this elaborate electronic hocus-pocus is worth the effort and the resources. Although personal computers clearly offer substantial services to a single user, is the local network concept viable?

"A personal computer will satisfy your needs 90 percent of the time," answers David, "but that other 10 percent is very important; it's not just 10 percent of the problem, and there's no alternative to getting it. Communication with other computers and users is ab-

Continued on page 4

XERPT

Published by Xerox Corporation, Printing Systems Division. Address queries to XERPT Editor, 701 So. Aviation Blvd., MS-A1-10, El Segundo, CA 90245. XEROX

OFF-LINE

The Acronym Game (TAG) Continued (C)

In our last issue (which was also our first), XERPT introduced The Acronym Game (TAG). It wasn't much of a game, of course, because the answers were printed, upside down, in the same column. It wasn't too much fun for us, either, because we have no idea how many people tested their acronym skills.

We think we've come up with a more appealing idea, so here we go again: XERPT introduces The Acronym Game Contest (TAGC).

There will be three winners, each of whom will receive a Cross Pen and Pencil Set (nice), as follows:

1. To the person who submits the greatest number of original acronyms.
2. To the person who submits the most outrageous single acronym in the real world.
3. To the person who submits the most original acronym not necessarily in the real world.

The decisions of XERPT editors will be final. Xerox employees are not eligible to win under the infamous XEANETW rule (Xerox employees are not eligible to win). Entries must be postmarked by October 15, 1979, and should be sent to XERPT Contest, Dep't. A1-10, 701 So. Aviation Blvd., El Segundo, CA. 90245.

Tracking Our Tracts

Our lead story in the last issue dealt with the then upcoming auction of oil lease tracts within the federally controlled part of the outer continental shelf (OCS) located between California's Santa Barbara Channel and San Pedro Bay.

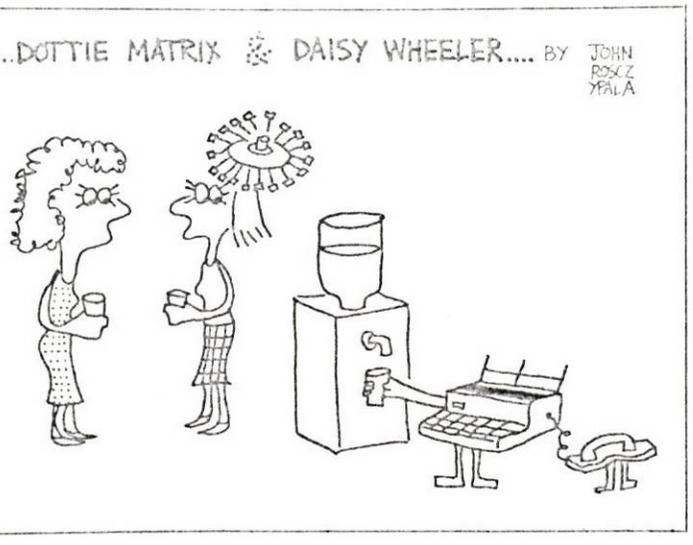
Oil lease bids totalling $572,825,417.90 covering 54 tracts were accepted by the U.S. Interior Department, according to the Bureau of Land Management, which conducted the June 29 sale. Because that is a rather substantial amount of money (well in excess of XERPT's budget), we thought readers would appreciate a few details in case they missed the story in the newspapers.

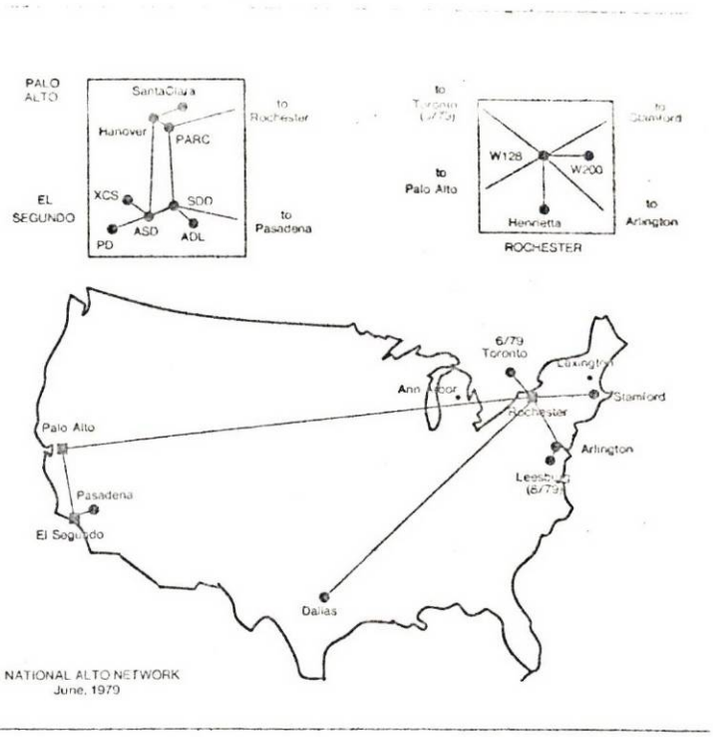
A total of 148 tracts (792,845 acres) were offered under sealed competitive bidding procedures conducted in Los Angeles. Fifty-five of the tracts received bids. Only one of the high bids received on the 55 tracts was rejected, based on U.S. Geological Survey's suggested minimal bid price.

The highest bid received per acre was $37,280.00, the third highest bid in national OCS lease sales. The highest bid received on a single tract was $95,475,000.00, covering an area located about six miles off Pt. Conception.

Daisy Pickin' Time

"In" customers will recognize this quaint name as referring to a lease to purchase conversion program that allows *free* rental credits for customers who purchase their leased Xerox terminals before October 31, 1979. The bonus, equal to three months of additional rental credit, is at the current percentage rate and is available even if the maximum accrual amount has been attained. The program covers any customer currently leasing a Xerox 17XX terminal.





Xerox's internet network reveals gateways (dots) and California and Rochester area interconnections (see insets). At least one Ethernet network (with from 2 to 256 hosts) is attached to each gateway. Lines are 9.6 Kbit leased phone lines.

Ethernet Network

Continued from page 3

solutely essential."

The concept of providing communications among personal computers followed implementation of the U.S. Defense Department's *Arpanet*, a packet communications network linking large-scale time sharing machines, and the Aloha Network at the University of Hawaii, which applied packet radio techniques for communications between a central computer and its terminals dispersed among the Hawaiian Islands.

The Ethernet concept, however, was the first major local network experiment to achieve widespread recognition in the scientific community. In recent years this technology has spawned new configurations as well as additional public and private networks. □

Ethernet Network Glossary

ALTO: The name of a minicomputer designed at PARC in 1973 and used extensively as a host (or station) throughout the Ethernet system. The standard Alto computer includes an 875 line raster-scanned display; a keyboard, including a "mouse" pointing device, and a five-finger keyset; a 2.5 Mbyte cartridge disk file; an interface to the Ethernet system with a 3 Mbit/second communication facility; a microprogrammed processor that controls input-output devices and supports emulators for a number of instruction sets; and 64K 16-bit words of semiconductor memory, expandable to 256K words.

ARPANET: The Defense Department's Advanced Research Project Agency (ARPA) is the sponsor of the Arpanet packet communications network which, in 1979, included 65 sites and approximately 120 computers. The communications subnet is composed of interface message processors (IMP) and terminal IMPS linked in a distributed fashion by 50,000 baud synchronous wide band circuits.

FILE SERVER: A computer that provides a file storage and retrieval service.

GATEWAYS: Computers that connect separate Ethernet networks. They may also perform certain server functions such as mapping a string of characters to a *network address* (see below).

INTERNET: A network of networks; an *internet* provides the ability for any two hosts to communicate if their local networks are interconnected.

MAIL SERVER: Like a mailbox. A computer that stores message files for an Ethernet network's electronic mail system.

NETWORK ADDRESS: A user's network and host numbers that uniquely identify any station on the *internet*.

"PUP": The name of an internet packet format, a hierarchy of protocols and a style of internetwork communication created at PARC.

IFIP 6.5
Syst.Env.Group
European Section

N77

European SEG meeting report on Names, Directories and Lists

BONN Oct.82

ROME Jan.Feb.83

Abstract

After a first meeting in Bonn, St. Augustin, hosted by GMD, where most of the discussions were concentrated on Naming, Naming Domains, Directory Service and Distribution, another meeting has been organized in Rome (January 31 to Februry 2 1983) hosted by IMB-Italy. The same points have been discussed extensively. The result is presented in this document.

The editing of this document has been made by the chairman of the group who assembled the different remarks and suggestions made by P. Dell'Orco, M.T. Pazienza, G. Schulze and F. Sztajnkrycer who have prepared separate contributions.

The ideas contained in this document are to be considered during the April workshop in Palo Alto, as possible contribution to the final living document on Naming, Addressing and Directory Services for MHS, which is under preparation by the whole W.G.6.5.2.

I. INTRODUCTION.

In the context of international mail system composed of private CBMS, public CBMS, used by a variety of clients having different sets of capabilities and belonging to different groups (humans, machines, ...), the problem of Names and Management of Names is crucial to the good functionning of the global system.

Problems related to names are :

- assign unique names to clients ;
- process efficiently the queries about the clients ;
- clients may have a very rich set of properties that they wish to include in their naming schemes.

In our paper, we designate the client wishing to use the mail system as the Communicating Entity (CE). The selection of this generic expression allows us to designate a process, a machine, a human being or an organization without any confusion.

On the other hand, since the word Name is used very often in the communication area, we will introduce the word Identifier to designate the unique reference that is managed by the mail system, to identify the client or the Communicating Entity.

In this report, we propose a strategy for identifying CEs, and for creating domains to manage correctly the CEs identifiers. A model of directory service is also described. Finally, we give our view on the Distribution List definition, mode of operation and the expansion problem.

2. WHAT IS A COMMUNICATING ENTITY ?

In an international Mail System, a Communicating Entity can take different forms :

Individuals :

- human being
- role
- organization
- process
- computer resource

Collections :

- distribution lists
- groups

A reliable mail system will have to deliver mail to these CEs without any ambiguity. In order to achieve this, a basic requirement consists of assigning a Unique Identifier to each CE.

In the global message system, as defined in the model which consists of a set of cooperating MTAs (message transfer agents) and a set of UA (user agents), entities that communicate exist also in order to achieve a particular job (transferring a message or a protocol data-unit). However, these entities are considered as internal system elements and do not belong to th class of Communicating Entities which use the mail system to exchange mail items.

The distinction between Individual Communicating Entities and Collective Communicating Entities is very important because of the processing during message transfer and localization that is attached to each category :

> An Individual CE is recognized as such and leads to the delivery of one message. A Collective CE group or distribution list leads to multiple message delivery. The members of groups, or distributions, may be Individual CEs and other groups or distrubution lists, the expansion of which leads to Individual CEs. Care should be taken during multi-step expansion (as we will see in a later section) to avoid looping when lists embed other lists.

3. IDENTIFICATION SCHEME.

Considering the fact that communication can be based on different criteria : personal, professional, social, etc..., a universal mail system should provide a rich set of attributes to designate the CE. A combination of these attributes will produce a Unique Identifier for the CE. The analysis of the possible attributes that a CE may need has generated two sets of attributes each one divided into classes.

The first set deals with the traditional environment of the world, while the second set describes its technical capabilities. The separation into two sets is done only for explanation purpose and has no impact on the design.

The traditional set :

In this set, a very exhaustive list of properties may be defined. However, we limited ourselves to the most common characteristics which may be found in all countries and organizations. This should not exclude the addition of new international sets, by common agreements, or national and private options.

Personal_Class_:

Last Name, First Name, Initials, Qualifier, Telephone Number, Languages,....

Geographical_Class_:

Country, Sub-Country, Province, City, Street, Building, Floor, ...

Business/Organization Class :

Company, Department, Sector, Role, Function, Title, Telephone Number, ...

Professional_Relations_Class_:

Associations, Committees,....

Social_Class_:

Associations, Social Security N°, Unions, Political Parties, Taxe References, ...

The Technical Set :

This set reflects the new capabilities introduced by computer and telecommunication tools in offices and homes. Two classes have been identified :

Technical Personal Data Class : User-IDs, Passwords, Mailboxes-IDs, Distibution Lists, Protection criteria X121 Address, Telephone Number, ...

Technical I/O Capabilities Class : Terminal Types, Protocols Supported, Emulation Modes, ...

REMARKS : Although this list of classes is oriented toward human users, it can cover the case of processes and computer resources with the technical classes. So it can be considered as a superset for a worldwide space of Communicating Entities.

Now that we have a variety of attribute sets, it is easy to create from specific combinations of attributes Unique Identifiers to a Communicating Entity (UI-CE). Every Communicating Entity will be identified with a set of Attributes, each one being represented by a type-value pair.

For instance :

UI-CE : (Last Name = MARTIN, First Name = Jacques, Qualifier = Prince, Languages = English, French, German)

UI-CE : (Country = France, Département = Seine, Town = Paris, Street = Champs Elysées, Street Number = 88, Last Name = Raspoutine)

UI-CE (X121 address = 0311061700138)

UI-CE : (Political Party = Liberal, Reference = 6868)

The most important condition to satisfy, in order to reach a unique CE without ambiguity, is to have a control over each attribute type and all possible values it may take. There should be an authority which assigns attribute types and defines the range of values authorized for each attribute.

There may be different combinations of attributes that lead to the same Communicating Entity. But, there is no well formed combination that leads to more than one Communicating Entity whether it is individual or Collective. Here is an example illustrating the first case :

CE : (International Organization = Red Cross, Area = Northern France, Role = Treasurer, Last Name = Dupont)

CE : (Country = France, City = Paris, Street = Lalo, Last Name = Dupont).

The mail system will keep independent these two sets and does not try to avoid any duplication in transferring twice the same message to the same CE. On the other hand, for the destination

CE, the path carries some semantics which should be preserved by the message system.

In addition we will see that, with the introduction of domains and subdomains for managing Name Spaces, it is impossible to keep track of all duplications.

Arbitrary combinations leading to a unique CE can be used by a sender. This will be provided by the Directory Service offering sophisticated facilities of Data Base Management System on attributes. When attributes, given by the sender, are not sufficient to identify one unique Individual CE, either one of the two following solutions may be selected : the Mail System delivers to the first CE satisfying the criteria, or it informs the sending CE about the existance of ambiguity without telling it about the nature of this ambiguity (privacy reasons). From this point, the sending CE may decide to continue refining the selection and gives more attributes or asks the system to stop the interaction (these problems will be presented in the Directory Service section).

4. NAMING DOMAINS.

The multiplicity of descriptions that may be assigned to or requested by a CE, makes it necessary to introduce the concept of Domains, which may be divided into sub-Domains, whose basic goal is to act as an authority for managing a limited space of attributes.

Management functions include :

- creating or deleting an attribute type ;
- creating, deleting and modifying attribute values ;
- controlling the uniqueness of each attribute type and value pair.

The world organization and the dispersed geographical sites where computer resources and human users can be located, will promote a hierarchical structure of well limited areas or domains. A tree-like representation of the whole naming scheme, based on attribute sets as seen previously, will be the most natural (comprehensive to the human user) and the most easy to implement in terms of distributed computing systems.

The proposal is to start from a common root qualified by an attribute type = World leading to the highest level domain. This root will be responsible of a set of attributes such as Country, International Organization. It will manage all its sons as indicated above. No new country is added without the authorization of the World Domain. Changing Country value = France to Country value = Kingdom of France, necessitates a specific request to the Domain Manager. In International Organizations we may find attribute values such as Red Cross, IUT, International Systems n, ...

Every Node, issued from the World Root, may be divided and give birth to a tree of nodes (Country, Region, Cities, ...) or to a flat space of nodes / leaves (in an International System n we may have all Communicating Entities identified by their Universal Reference Number, [e.g. Ethernet Numbering Scheme]). The figure which follows shows a representation of such a tree.

There is a distinction between Naming Domains, as defined above, and Management Domains, as defined by the CCITT. Management Domains represent administrative boundaries between UAs, MTAs and groups of UAs and MTAs. While Naming Domains deal with the the management of attributes.

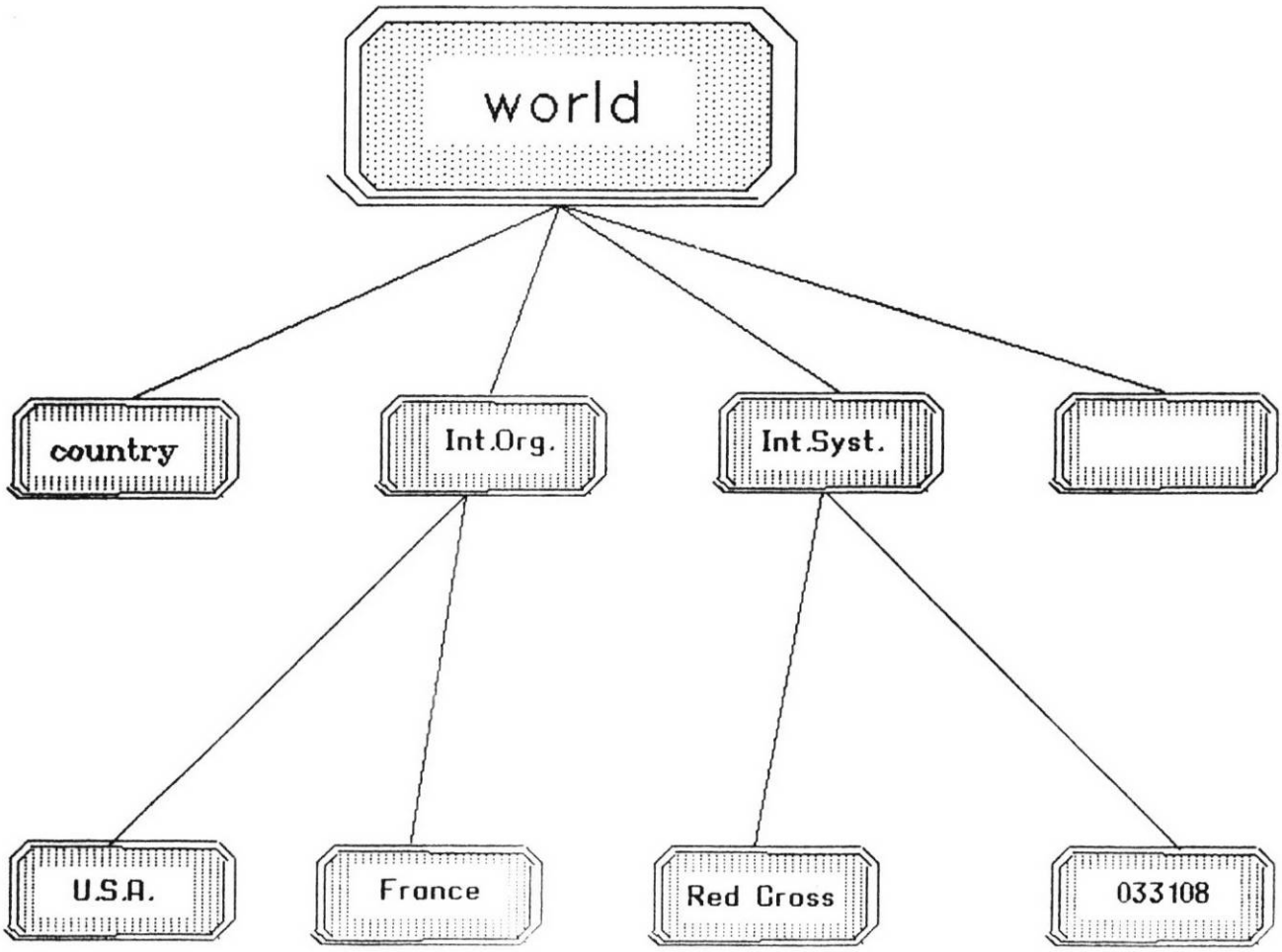


Fig.1 Naming Tree

As the tree shows, every parent node is responsible for creating the types of its children. For instance, the Country USA gives birth to children such as States and National Associations, while Country FRANCE creates children such as Départements and National Organizations. In order to designate a Communicating Entity in a deterministic way, every attribute type will be represented by a concatenation of the value of its parent and its own simple representation. For example, in a national organization such as IBM in France, there might be a Département attribute type, which will lead to a confusion with the attribute Département attached directly to France. For this reason, the second attribute type Département will be represented by "France Departement". (=Seine), while in the former case, département will be represented by "France.IBM.Departement" (=Marketing). With this scheme, every attribute type in the tree will have a unique syntax that indicates the path from the root.

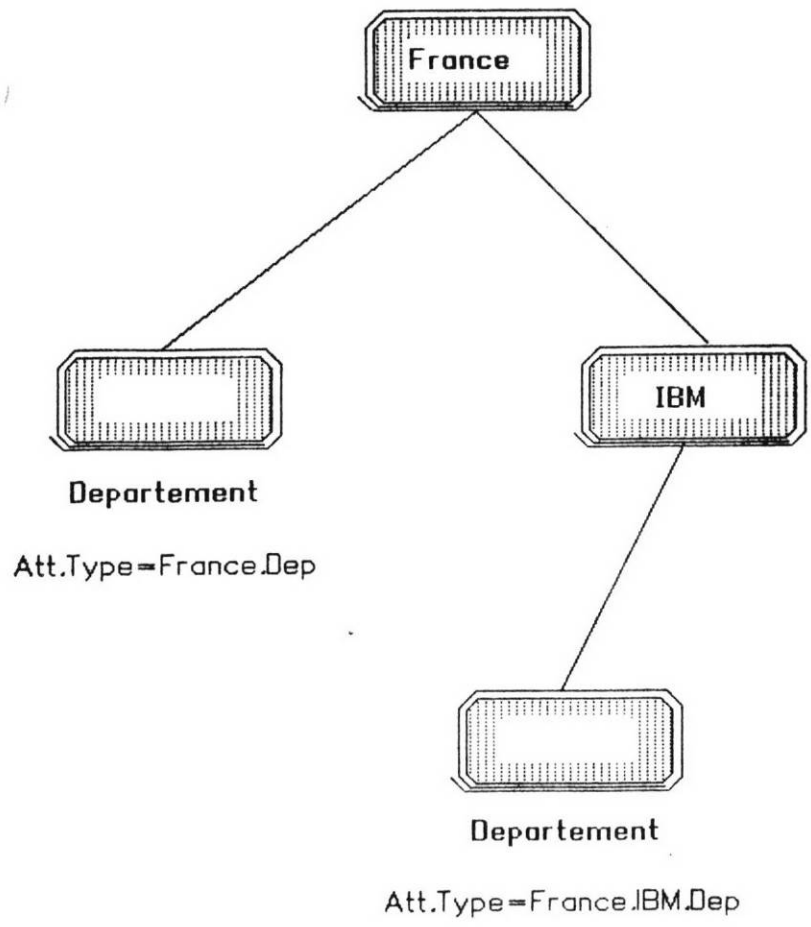


Figure 2

Domain names may occur more than once in the tree, they are only unique within the context of the father Domain. An example of duplicated names is illustrated by the following figure :

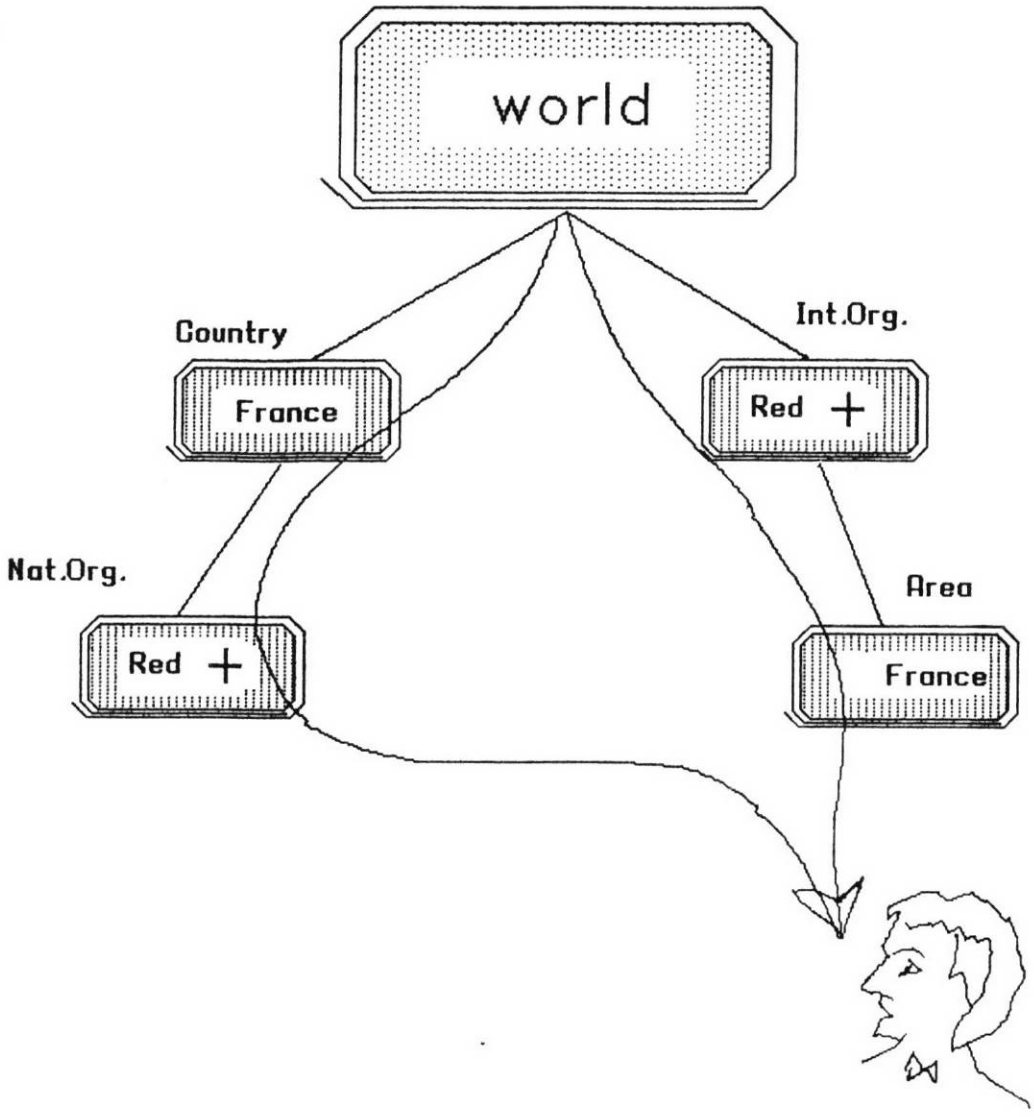


Figure 3.

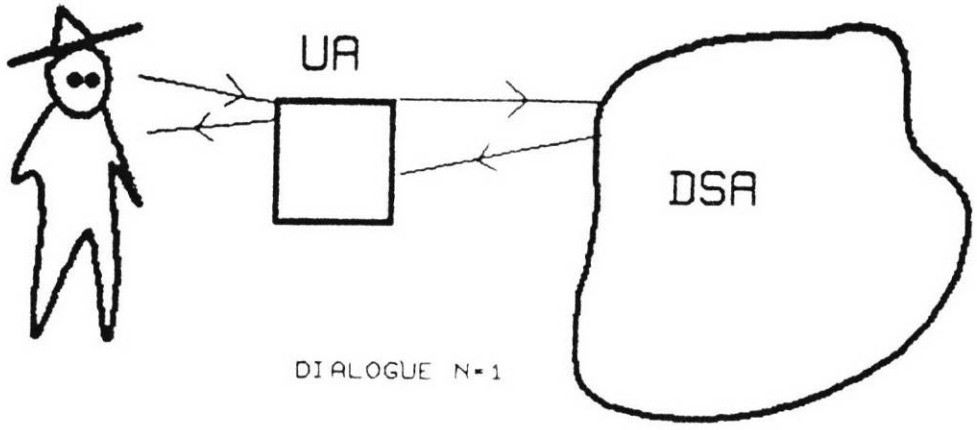
5. DIRECTORY SERVICE.

a) Functions Performed :

The directory service maintains a consistent database of attributes. Every attribute will be indicated with its type, value and a protection qualifier. The client requests for information by providing a particular attribute type. It gets back all the values authorized for such type. Example : for the request at the world level "Country ?", the client gets as answer all the country names representing all the values of Country registered in the World Domain.

Two categories of services are provided by the Directory Services : Information Retrieval type of service and Address Mapping type of service. Clients of the Directory Service may be UA's, MTA's and embedded agents of the directory service itself. The service should be reliable, and available.

Dialogue with directory service may be one of two types :



MTA=Message Transfer Agent
UA=User Agent
DSA=Directory Service Agent

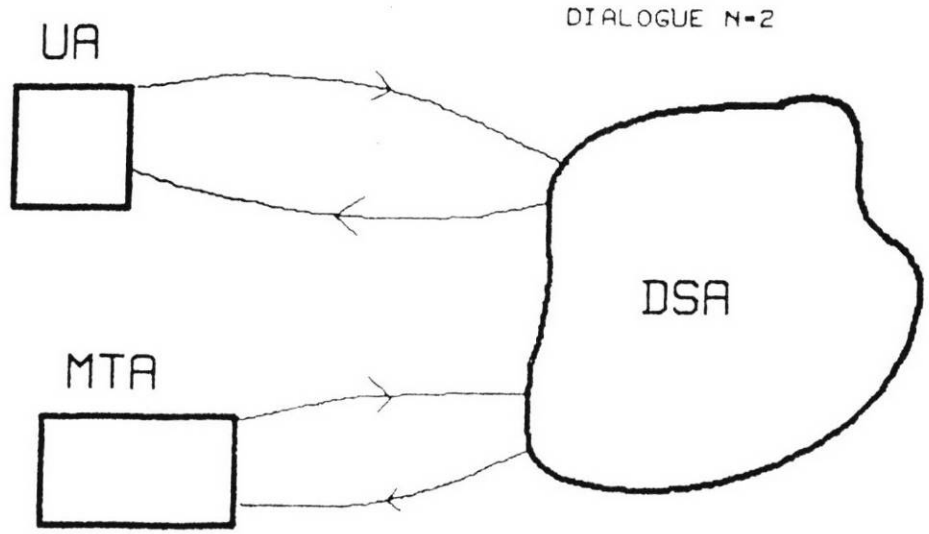


Figure 4 : Dialogue Types.

Dialogue [1] is a database interrogation dialogue where the user enters attributes. During this dialogue, which may take several iterations, the directory service performs spelling corrections and language translations : (e.g. Londres ⟶ London, Frankreich ⟶ France). The ultimate goal of the dialogue is to obtain from the directory service one (or several) correct CE(s) ; the associated address might also be communicated to the requestor.

Dialogue [2] is one-shot data-base interrogation where an attribute is supplied to the directory service, and an answer is given back which may be either negative or positive. The positive answer is accompanied by the associated address. The dialogues [1] and [2] differ mainly in their usage of the directory service.

It is possible that the database for dialogue [1] is extended towards an advertisement database and used as such besides the usage for the CBMS. The mapping of names or partial names to routes and the mapping of addresses or partial addresses to routes is explicitely excluded from the directory service. These mappings are performed via some MTS internal routing tables which is not relevant to this document.

b) Architectural model of the Directory Service.

The Directory Service is composed of a number of DSAs (Directory Service Agents). Some DSAs might maintain connections between them :

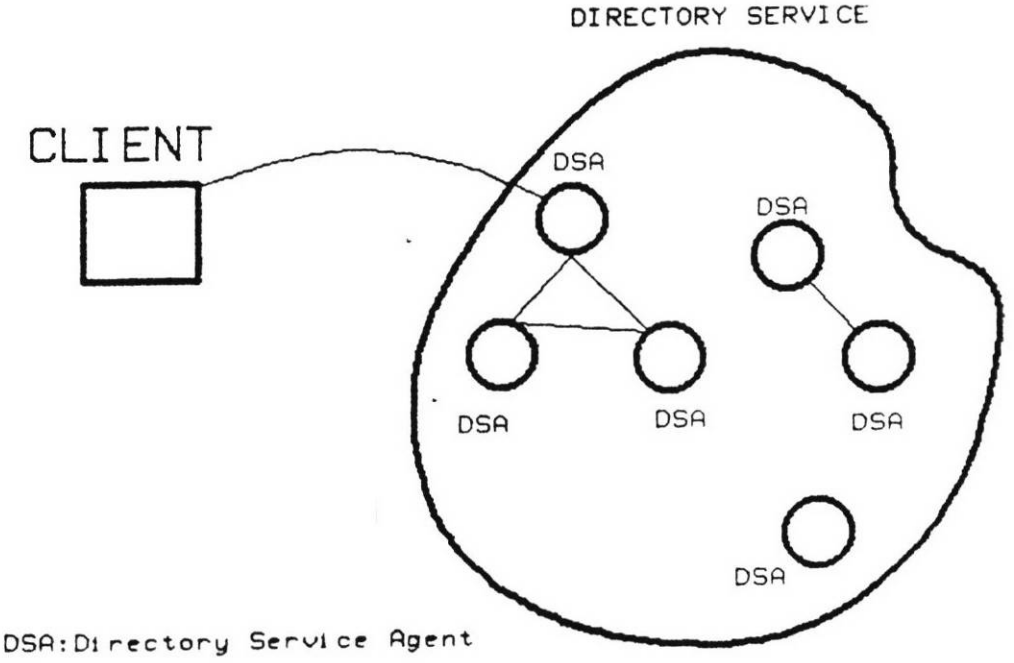


Figure 5 : Directory.

Communications between the clients (i.e. human users through their UA, UAs, and MSAs) and the DSA are based on a separate connection-oriented session established through a public data network. Such a connection may share the physical link with other CBMS traffic (UA⟺MSA) or other traffic that can be "CBMS-Independent".

It might happen that in some countries the CBMS is available before a public data network. In such case, the directory service traffic will utilize the connectionless transport of information through the CBMS. To every domain a DSA is associated. Several domains, as well as domains and sub-domains, may share a DSA. A DSA can perform dialogues [1] and [2] with respect at least to all domains and sub-domains to which the DSA is associated.

For any other domain known to the DSA, the DSA can provide the address of the associated DSA. This address might be used by the DSA itself which forwards the client's request directly (Fig. 4a) or the address is communicated back to the client, who has to persuade its request on his own (Fig. 4b). Both schemes can co-exist. In the case of a dialogue [1], the address returned by the DSA (step 2 in Fig. 4b) is intercepted by the UA, which re-issues the original request to the indicated USA, this operation might be disclosed to the human user.

Considering the possibility of a connectionless communication with some DSAs, a "quality of service" indicator is returned together with the address of that DSA. Client action might differ for different classes of "quality of service". Especially, the UA intercepting these addresss can behave differently.

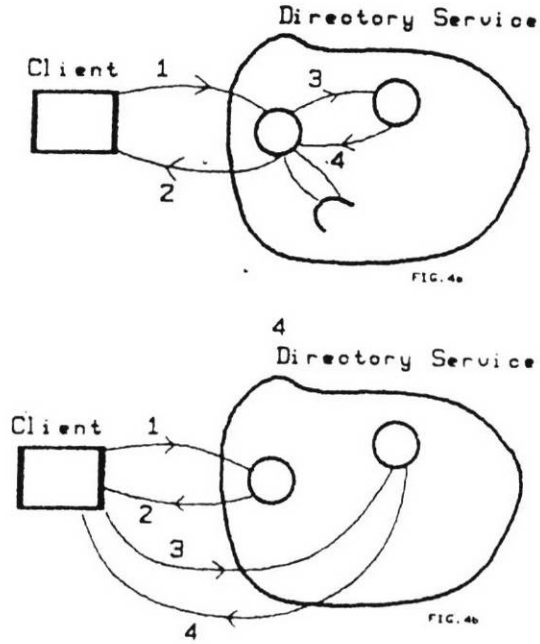


Figure 6 : Scenarios.

6. DISTRIBUTION LISTS.

6.1 Definition.

A distribution list is a Communicating Entity in the mail system that can be identified with an appropriate set of attributes. The purpose of a distribution list is to facilitate the sending of mail items to a group of recipients and the same time to enable the MTS to economize on communication resources, i.e. the originator has only to state the distribution list as a recipient of his mail item and the MTS may produce the individual copies of the mail item for the different recipients at the latest possible stage in the transfer.

The distribution list is a set of members, each member represented by a set of attributes and a set of properties. The attributes follow the scheme defined previously. The properties of the member define the access and processing capabilities of the entity described by the set of attributes as well as properties like "financial sponsor".

In addition, each distribution list contains a set of general attributes describing the operation which may be done on the list by other CE who are not necessarily members of the list. In addition, procedures may be attached to perform a particular processing around the list, eg. : Teleconferencing, patterns of mail circulation and processing.

There is an important distinction to make between a Group and a Distribution List. Both of them are Collective Communicating Entities. But the Group does not necessitate a specific request from a Client to be created. While a Distribution List is created explicitely after a request sent by one or more CEs. Thus, a Group is attached implicitely to any intermediate node of the naming tree. Members of the Group are their children which may represent other Groups or Individual CEs. Sending a message to a Group can be done simply by giving the value of the attribute representing an intermediate node (e.g. : send Message To Country = France), and the message will be forwarded through all the paths shown in the figure below. There is no control of duplication of message delivered to the same individual CE.

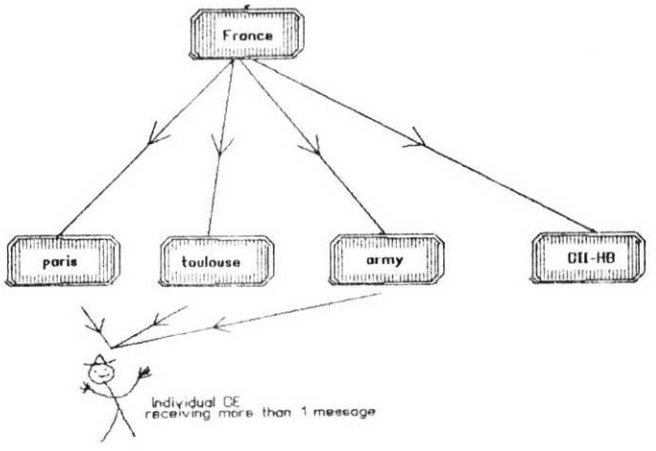


Figure 7.

A Distribution List has more semantics in it than a reduced addressing as in the case of a Group. It may be created for business purpose, and it may have particular properties that reflect the role of each member of the list. The other difference is that members of the List can be constituted of other Lists, individual CE or groups belonging to different level of the Tree (e.g. : List X is composed of USA, France.Air Force, France.Paris.Champs Elysées.TOTO).

The expansion of Lists (indentifying its members from its name) is performed through the Directory Service data-base containing members' names and addresses.

In order to have unique identifiers for Lists created all over the world, we follow the scheme of naming proposed previously. The CE requesting the creation of the List will address its request to the Naming Domain Manager to which it is attached. In the List name, will appear the complete path, leading to the Naming Domain (remember that it is a unique path), plus a List type to which we give unique values that make the distinction between all Lists created at this level.

6.2 Operations on Distribution Lists.

Creation : every CE, or group of CEs, is/are able to create a distribution list. During the creation process, they give sets of attributes to this list, and every set of attributes is checked (ex: a member cannot be introduced, if we find that he indicates in his technical personal data class that he does not want to be on distribution lists).

Properties which may be assigned to members of the list by its creators are :

capability to
- modify/alter membership
- delete member and/or property
- add member and /or property
- list member and/or properties
- read selectively member, properties.

6.3 Expansion of Distribution Lists.

There are two message transfer related operations in connection with distribution lists :

1. expansion of distribution list name, and
2. multiplying the message and sending it in different directions. These two operations are separate and don't necessarily have to be performed in the same MSA.

An example illustrates a possible mechanism : assume that an originator in, say, Austria sends a mail item to the recipient with the name "(List A) (Zurich) (CH)", we assume in this example that the set of attributes Name, City, Country, represent the list. The mail item is forwarded to MSAs in Switzerland. Eventually, the mail item reaches an MSA where an address is required and the MSA undertakes a type 2 dialogue with the directory service.

The answer from the directory service is positive but instead of an address the expansion of the distribution list is returned (note : this is an ammendment to the definition of the type 2 dialogue). This MSA has thus expanded the name of the distribution list. The recipient field in the envelope is modified to look like :

(List A) (Zurich) (CH) - done / (Name 1) (Paris) (F)/ (Name 2) (Bonn) (D) / (List B) (Paris) (F)

indicates that the particular recipient field has been treated.

The MSA now makes a routing decision for all un-treated recipients. Let us assume that Switzerland maintains a "gateway" - MSA ; thus, the mail item must be forwarded to this gateway - MSA for all recipients. In this case, the MSA that has just performed the expansion does not multiply the mail item ; it sends a simple copy to the gateway-MSA. The gateway-MSA makes again a routing decision for all untreated recipients. Now, two routes emerge, one towards Germany and one towards France. Therefore, this gateway-MSA multiplies the mail item and sends one to an MSA in Germany with the recipient list :

"(List A) (Zurich) (CH) - done /(Name 1) (Paris) (F) - done / (Name 2) (Bonn) (D) / (List B) (Paris) (F) - done / " and to an MSA in France with the recipient list : "(List A) (Zurich) (CH) - done / (Name 2) (Paris) (F) (Name 2) (Bonn) (D) - done / (List B) (Paris) (F)".

The mail item sent to France will eventually reach an MSA in the Paris area where again a name to address translation is required. The response from the directory service will be an address (for name 2) and the expansion of list B. The expansion is again included in the recipeint field which now looks as follows :

"(List 1) (Zurich) (CH) - done / (Name 1) (Paris) (F) - address / (Name 2) (Bonn) (D) - done / (List B) (Paris) (F) - done / (Name 3) (Paris) (F) / (Name 2) (Bonn) (D) / (List A) (Zurich) (CH)". [i.e., we have a case of recursive list references].

However, because the last two recipient names on the envelope appear already on the recipient list, these names can be deleted from the list or tagged as "done".

This algorithm has the following properties :

- the MSAs need no knowledge that a name designates a distribution list ;

- recursive list references are detected and no infinite loops can occur ;

- it cannot be avoided that a recipient receives the mail item more than once (e.g. if the name : (Name 2) (Bonn) (D) designates a distribution list that contains the member (Name 3) (Paris) (F).

Complete avoidance of the multiple delivery can only be achieved if all expansions are done at one place, in which case the MSAO is the most likely candidate where this expansion could be done. On the other hand, the receiving UA might detect such a duplication by comparing the mail item identifiers.

LIST OF PARTICIPANTS

** IFIP WG 6.5 SE.GROUP.EUROPEAN.SECTION **

Bonn Meeting - Oct. 26-28 1982

DZIDA Wolfgang	G.M.D. - Bonn - WEST GERMANY
KARMOUCH Ahmed	KAYAK Project, INRIA - FRANCE
KILLE Steve	U.C.L. - London - U.K.
NAFFAH Najah	KAYAK Project, INRIA - FRANCE
PANKOKE-BABATS Uta	G.M.D. - Bonn - WEST GERMANY
PAZIENZA Maria Teresa	Univ. of Bari - ITALY
SANTO Horst	G.M.D. - St. Augustin - WEST GERMANY
SCHICKER Peter	Zellweger Uster AG - SWITZERLAND
SCHULZE Günter	G.M.D. - Bonn - WEST GERMANY
SMAALAND Knut	Univ. of Oslo - NORWAY
SZTAJNKRYCER Fred	KAYAK Project, INRIA - FRANCE
WOSNITZA Lothar	Univ. of Düssefdorf - WEST GERMANY

Rome Meeting - Jan. 31 to Febr. 2 1983.

KARMOUCH Ahmed	KAYAK Project, INRIA - FRANCE
NAFFAH Najah	KAYAK Project, INRIA - FRANCE
Dell'ORCO P.	IBM - Rome - ITALY
PAZIENZA Maria Teresa	Univ. of Bari - ITALY
SANTO Horst	G.M.D. - St. Augustin - WEST GERMANY
SCHICKER Peter	Zellweger Uster AG - SWITZERLAND
SCHULZE Günter	G.M.D. - Darmstadt - WEST GERMANY
SZTAJNKRYCER Fred	KAYAK Project, INRIA - FRANCE
ZSCHINTZSCH H.	Univ. of Düsseldorf - WEST GERMANY

OAD 6-NOV-80 75437

Francine Field

THE EXECUTIVE MAIL SYSTEM MANUAL

TABLE OF CONTENTS

INTRODUCTION

MAIL is an Executive program for reading and sending messages, and for manipulating the messages you receive. The files in which the messages are stored have a "message file format". Commands in MAIL are initiated by typing their first character, which causes the program to type out the rest of the command word and wait for your input.

Before the commands are described, there are a few general statements about how MAIL works and some conventions used in describing the commands that you should know about.

When MAIL is called (by typing "mail<RET>" at the Executive herald "@") it will first try to read the MESSAGE.TXT file in your connected directory. If this file does not exist, MAIL will tell you so. If you are not connected to your login directory, MAIL will then try to find a MESSAGE.TXT file there. If that also fails, it will tell you so and wait for a command to be typed. If it finds a MESSAGE.TXT file, MAIL will scan it and display the header information (i.e., the Date, From, and Subject fields) for each message since the file was last read, preceded by a message number sequentially assigned by MAIL. These message numbers are used with the various commands.

If MAIL is called by typing "mail<SP>" at the "@", it will ask you for the file to be read. Typing <ESC> (an ESCAPE) requests MESSAGE.TXT to be read; confirmation (with <RET>) will be requested from the user. Once a file name has been specified and acknowledged, then the same information as described in the previous paragraph will be displayed.

When you ask for a message file to be read in MAIL, either when first calling MAIL or with the Read command described below, the file must be in the so-called "message file format". If MAIL recognizes that the file does NOT conform to this format, it will tell you so.

The herald letting you know that MAIL is waiting for your command is "<-".

There are five types of input that MAIL expects:

(1) a MAIL command (or subcommand) character
(2) a message sequence specification
(3) a file name
(4) a confirmation character (<RET>)
(5) a user name and/or remote site name

If MAIL does not understand your input, it will return to the herald or reprompt you. To abort MAIL's printing on your terminal, type <CTRL-O>. Many commands can be aborted with <CTRL-N>.

In this manual, <CTRL- > surrounding a character means to type the character while holding down the CTRL key. < > surrounding a capitalized word denotes other special characters, such as <RET> meaning the "return" key.

"FILENAME" is used in this manual to indicate the name of a sequential file. Enter the name in Executive form, terminate with <RET>, and when asked for confirmation (e.g., [Old version]) type another <RET>. You may include a directory name if needed (e.g., <DIRECTORY>FILE.EXTENSION;NUMBER), and <ESC> to fill it out.

MESSAGE SEQUENCE -

Specification of a message sequence, indicated by MESSAGE-SEQUENCE in this manual, is prompted by "(message sequence)" when Long mode is set. (See the Verbose command under "Other Commaods".) Select one of the following:

(1) Any single message number.

(2) Any two numbers separated by ">" or ":". This means a number sequence indicated by the two outside numbers (e.g., 2>5 or 2:5 means messages 2, 3, 4, and 5 in that order). If the first number is greater than the second number, it means the sequence in reverse order (e.g., 5>2 or 5:2 means messages 5, 4, 3, and 2 in that order).

(3) A pair of numbers separated by "-". The interpretation of the string "21-4" is message numbers 21, 22, 23, and 24. The string "24-1" is an error.

(4) Any sequence of the previous three types separated by commas. This is the way to group several nonadjacent messages together. For example, "1,3,5:7,10" means messages 1, 3, 5 through 7, and 10.

NOTE: MESSAGE-SEQUENCEs of the types described above are always terminated by <RET>.

(5) There are special types of message sequences that are recognized by one character. They are:

<ESC> -- causing the current message number to be echoed and the relevant process performed on that message only.

<CTRL-I> -- causing the previous completely specified MESSAGE-SEQUENCE to be echoed and processing performed on those messages.

R -- standing for "Recent" messages only.

O -- standing for "Old" messages only.

A -- standing for "All" messages.

D -- standing for "Deleted" messages. It may be used in the Headers and Undelete commands. It has no effect in the Delete command. It may be used in all other commands allowing MESSAGE-SEQUENCE; the headers of the deleted messages will be displayed but the command will have no effect on them.

U -- standing for "Undeleted" messages, i.e., messages that are not deleted.

I -- standing for messages in "Inverse" order. It prints the messages in the opposite order from the A (All) subcommand.

S -- standing for "Subject field search for string", which asks you to provide a string that will be searched for in the Subject field of the message headers, and selects those messages where it's found.

F -- standing for "From field search for string", which is like S but searches for the string you specify in the From (author) field of the message headers instead. When a message was sent by the logged-in user, the initial part of the "To" list of the message (if it exists) appears in the header in place of the author's name. Therefore, to search for messages from yourself, specify the string "To:" rather than your login directory name.

NOTE: S (Subject) and F (From) require you to type a string terminated by <RET>. The search will be performed on the string you type up to (but not including) the <RET>. The string you type must exactly match some substring of the appropriate field, except that the capitalization of alphabetic characters is ignored, and returns in the Subject field are ignored. If you do not type any string before typing <RET>, no searching will be performed.

E -- standing for "Examined" messages, i.e., all messages that have been completely typed (with the Type command) or listed (with the List command).

N -- standing for "Not examined" messages, which is the opposite of the E (Examined) subcommand.

L -- standing for the "Last" message sequence that was completely specified.

MAIL COMMANDS

For clarity in the command format below, what MAIL displays is in lowercase and what you type is in uppercase (except where FILENAME is used for the particular name MAIL displays), although this is not the case when using MAIL.

Commands to Send Messages

<- Send [confirm] <RET>

This command starts the SEND process. See "The SEND Process" below. If you type <CTRL-N>, you will be asked if you wish to abort. If you type <RET>, you will be returned to the top level of MAIL; otherwise, you will be allowed to continue with SEND.

<- Answer message number: MESSAGE-NUMBER
reply to those whom the message is: ANSWER-SUBCOMMAND
additional addresses (? for help):

This command allows you to send a message back to the sender of a message, as well as (at your discretion) to those people to whom that message was sent, without having to type their addresses.

MESSAGE-NUMBER can be one of the following:

any message number, terminated by <RET>

E -- for the "End" of the messages

L -- for the "Last" message (same as E)

B -- for the "Beginning" of the messages (message number 1)

<ESC> -- for the current message number

<RET> -- for the current message number

ANSWER-SUBCOMMAND can be one of the following (typing anything else aborts the command):

F -- for "From", indicating the sender of the message only

T -- for "To", indicating the sender of the message and those addresses in the To list

C -- for "Cc", indicating all recipients of the original message in addition to the sender of the message

A -- for "All" (same as C)

<RET> -- (same as F)

The header of the message specified is displayed so that you may be sure you are answering the correct message. It is displayed after you have specified the message number, but before you are asked to supply the subcommand.

When prompted for additional addresses, any that you specify will be included in the Cc list. (See "The SEND Process" below.)

An attempt is made to ensure that all addresses are valid (i.e., all host names for remote addresses, and user names in local addresses), and that no duplications are present. If clarification from the user is necessary, you may be asked some questions; in this case, all type-ahead is deleted.

When you are asked to complete the message body, control is given to the SEND process, as if you were prompted with "Message (? for help):". (See "The SEND Process" below.) Type your answer. If you type <CTRL-N>, you will be asked if you wish to abort. If you then type <RET>, you will be returned to the top level of MAIL; otherwise, you will be allowed to continue with SEND.

There are two relevant profile modes. See the Zap profile command under "Other Commands". One mode sets whether you receive copies of the answers you send. It is initially set for automatic receipt. If you have specified in the Zap command that you are not to receive copies of the replies you send, then your name will not appear in the Cc list unless you specify it as an additional address.

Another profile mode sets whether your replies are saved in a file named SAVED.MESSAGES. If you have specified in the Zap command that the automatic copies of your replies are to be stored in SAVED.MESSAGES, and if that file exists, copies will go into that file and not into your MESSAGE.TXT file. If you have not specified that copies are to be stored in SAVED.MESSAGES, if you are automatically receiving copies, and if there is a file named SAVED.MESSAGES in your login directory, you will be asked by the Answer command whether you want your copy to go to that file. If you confirm with <RET>, then your name will not appear in the Cc list.

<- Forward (message sequence) MESSAGE-SEQUENCE

This command allows you to send copies of messages you have received to other people. The headers of the messages you specify are displayed. Then you are asked for the subject of the forwarded messages; type a title or short summary and terminate with <RET>. Then control is given to the SEND process, where you are allowed to add your own comments or edit the message(s). When done, type <CTRL-Z> and then specify, in the standard way, to whom the mail is to be sent. Continue with the final step of "The SEND Process" described below. If you type <CTRL-N>, you will be asked if you wish to abort; if you then type <RET>, you will be returned to the top level of MAIL.

The SEND Process

The SEND process can be started in MAIL with the Send command, as indicated above, or by typing "send<RET>" at the "@" in the Executive. You will be prompted for each step as shown in quotes below.

Use the following control characters in composing and editing:

<CTRL-A> -- deletes the last character you typed
<CTRL-B> -- inserts the following file
<CTRL-H> -- same as <CTRL-A>
<CTRL-O> -- stops typeout
<CTRL-Q> -- deletes the last line you typed
<CTRL-R> -- re-displays the current line or step
<CTRL-S> -- re-displays the entire text or all steps
<CTRL-W> -- deletes the last word
<CTRL-X> -- cancels the current step (ignored during text input)
<CTRL-Z> -- terminates the message

"To (? for help):"

Enter the user names in the form NAME1, NAME2, ... and terminate the list with <RET>. For addressees at other hosts on the ARPANET, type the user name in the form NAME@HOST-NAME. Separate addresses with a comma, and terminate with <RET>. Typing @HOST-NAME as a single address causes that host to apply to subsequent names in the list.

Distribution lists are a convenient way of specifying addresses. A sequential file containing addresses in the form described above can be made, for example, a file named "hq-staff.list;1" containing "Jones, Smythe@Office-1, Williams". If such a file is available, type <CTRL-B> and specify the FILENAME, for example, "<CTRL-B>hq-staff.list;1<RET><RET>". The response "...eof" (end of file) means the list has been taken. It works just as if you had typed the names addresses yourself.

Each message a user receives indicates to whom else the message was sent, even if the names came from a distribution list file. That is not the case when the sender includes some text ending with a colon in front of a list of addresses, whether or not it came from a distribution list file. In this case, the text and colon will appear in the message instead of the names that followed it. For example, if "Headquarters Staff: Jones, Smythe@Office-1, Williams" is specified by the sender, only the text "Headquarters Staff:" will appear in the resulting message.

To send the message to a file other than someone's MESSAGE.TXT file, specify the FILENAME preceded by an asterisk as an address, for example, "*correspondence.txt;1".

"cc (? for help):"

(Same as for "To" above)

"Subject:"

Type a one-line title or summary of the message and end with <RET>.

"Message (? for help):"

Type your message. <RET> should be used to begin a new line. You can use the control characters listed above. A copy of the text of the message will be saved in your file MESSAGE.COPY;NUMBER;S for the duration of your login session. If you have a file named SAVED.MESSAGES in your directory, a copy of the message will also be written there for a more permanent record.

Terminate the message with <CTRL-Z>.

"Q,S,?,carriage-return:"

You may tell the SEND process either to try to send the message immediately or to hold (queue) it for later delivery and forwarding. You may choose from the following subcommands; end the series you type with <RET>.

Your basic choice is:

nothing -- (i.e., nothing but <RET>) selects the default - to send the message immediately if possible.

Q -- queues the message for all addresses except file names preceded by "*".

S -- sends the message immediately (the default).

S or Q may be immediately followed by:

L -- (local) applies only to local addresses.

N -- (network) applies only to non-local addresses.

S (and L or N if used) may also be followed by:

W -- (wait) tells how long to wait before sending the message. When prompted by "time=", enter a number of seconds. The process will wait that long for:

- -- a busy local mailbox to become free (default is don't wait)
- -- an ARPANET host to give permission to transmit (default is 30 seconds)

NOTE: If L or N also appears with S, the effect of W is restricted to local or network mail.

Commands to See and Manipulate Messages

<- Headers (message sequence) MESSAGE-SEQUENCE

This command displays the headers for the messages in the message sequence you specify. The headers of recent messages are preceded by a plus sign (+), messages that have not yet been examined (see the Type command) are preceded by a minus sign (-), and deleted messages are preceded by an asterisk (*). If a message was sent by the logged-in user, the initial part of the "To" list of the message (if it exists) will be displayed in the From (author) field of the header. The length of the message will be displayed along with the header if the I (Inclusion of length in header) command has been used.

<- Type (message sequence) MESSAGE-SEQUENCE

This command displays, or "types", the messages specified by MESSAGE-SEQUENCE. All messages that are completely displayed are treated as having been "examined".

<- Put (message sequence) MESSAGE-SEQUENCE into file name: FILENAME

This command copies the messages specified by MESSAGE-SEQUENCE into the file specified by FILENAME. If the file does not exist, it will be created, and the messages will be written into it. If the file already exists, the messages will be appended to the messages already in the file. Confirm with <RET>. This command is useful for keeping separate files containing messages concerning different topics.

<- Move (message sequence) MESSAGE-SEQUENCE into file name: FILENAME

This command is a convenient combination of the Put and Delete commands. As the messages are copied into the other file, they are marked for deletion in the current file. If any of the messages are already deleted, you will be told, and those messages will NOT be moved to the file.

<- Delete (message sequence) MESSAGE-SEQUENCE

This command classifies the messages you specify as "deleted". It marks each message with an asterisk preceding the header information, indicating that it is deleted. If you read the file again, the messages will still be marked and treated as deleted. Messages that are deleted are treated differently in later commands; for example, if you have deleted message number 5 and then try to Type or Put a message sequence including message 5, message 5 will NOT be included. Deleted messages will be expunged, i.e., permanently erased, when you give an Overwrite, Write, or Exit command.

<- Undelete (message sequence) MESSAGE-SEQUENCE

This command undoes the action of the Delete command for the messages you specify. They are no longer marked or treated differently than other messages which were not deleted. (Both the messages you undelete with this command and the messages that have never been deleted are referred to as "undeleted" messages.) Deleted messages that have been expunged, as with the Overwrite or Exit command, cannot be restored with any command.

<- List (message sequence) MESSAGE-SEQUENCE on file: FILENAME

This command lists all the messages specified by MESSAGE-SEQUENCE into the file specified by FILENAME, in a printing format rather than a MAIL-readable format. If the file does not exist, it will be created, and the messages will be written into it. If the file already exists, the messages will be appended to the messages already in the file. Confirm with <RET>. This command is intended to allow you to obtain a hardcopy listing of messages. All the specified messages are treated as having been examined. If you are listing more than one message, a preface page in the file will be created which contains the headers for those messages. You will be asked if you want each message on a separate page; if you do, confirm with <RET>; otherwise type "n" (for no). The preface page of headers may have the length of each message included depending on the setting of the Inclusion command. (See "Other Commands".)

Commands to Sequence through the Messages

<- Current message is NUMBER of NUMBER messages in file FILENAME

This command tells you the following:

-- the number of the current message
-- the total number of messages
-- the file name of the message file being read

The current message is the last message you examined. Or, if you have not examined one yet, it is the one before the first recent message; if the file has no recent messages, it is at the end of the file. Use this command to find out which message the Next and Backing up commands will display.

<- Go to message number: NUMBER

This will allow you to explicitly change the current message number. If NUMBER is not a valid message number (i.e., it is less than 1 or greater than the number of messages in the file), you will be told and the current message number will not be changed. However, there are several other alternatives for NUMBER:

E -- end of the messages (the last message)
L -- last message (same as E)
B -- beginning of the messages (message number 1)

<- Next message is:

This command displays the message following the current message (if one exists) and sets that message to be the current message. Deleted messages are not displayed, but the current message number is incremented. The displayed message is treated as having been "examined".

<- <LF> {the LINE FEED key}

Same as the Next command. It displays the message following the current message and sets that message to be the current message. The displayed message is treated as having been "examined".

<- Backing up -- previous message is:

This command displays the previous message (i.e., the current message number minus one) and sets that message to be the current message. It is the inverse of the Next command. The displayed message is treated as having been "examined".

<- ^ {uparrow}

Same as the Backing up command. It displays the previous message and sets that message to be the current message. The displayed message is treated as having been "examined".

<- <CTRL-H>

Same as the Backing up command. It displays the previous message and sets that message to be the current message. The displayed message is treated as having been "examined".

Commands to Update your Message Files and Leave MAIL

<- Overwrite old file FILENAME [confirm] <RET>

This command overwrites, or updates, the current message file (it echoes its FILENAME), expunging (permanently erasing) the deleted messages. For example, if you delete message 2 and then overwrite your file, message 2 will disappear. Overwrite also rereads your file, renumbering the messages. You will be warned if any unexamined but not deleted messages exist in the file that you are overwriting.

<- Write sorted file FILENAME [confirm] <RET>

This is similar to the Overwrite command, except that the messages are sorted into ascending sequence by their arrival time before the overwriting is done. The file is then reread. You will be warned if any unexamined but not deleted messages exist in the file you are sorting.

<- Quit [confirm] <RET>

This command returns you to the Executive without overwriting the file. It is similar to typing <CTRL-C>. You will be warned if any unexamined but not deleted messages exist in the current message file.

<- Exit and update old file FILENAME [confirm] <RET>

This command is another way to overwrite your current message file, but instead of rereading the file it returns you to the Executive. It is equivalent to doing Overwrite followed by Quit, but without the overhead of rereading the file. You will be warned if any unexamined but not deleted messages exist in the file.

<- <CTRL-E> exec [confirm] <RET>

When you type <CTRL-E>, you will see the word "Exec". Confirm with <RET> if you wish to temporarily enter a new instance of the Executive. When finished, return to MAIL by typing "pop<RET>". Your context in MAIL will be preserved.

Command to Read other Message Files

<- Read file name: FILENAME

You can use MAIL on any file that has a message format. This means you can peruse or modify files created with the Put or Move commands (but NOT the List command). If, for example, you have a file containing messages pertaining to MAIL problems, you can read it to make sure you have taken care of them. Read is the command that lets you read files other than your standard mailbox file, MESSAGE.TXT. It also prints out the recent header information for that file. If that file has old messages that have not yet been "examined", you will be told. You will also be told if any of the messages in the file are deleted.

Other Commands

<- Verbose

The amount of command text and prompting that MAIL displays for you is set to one of three modes: Long, Normal, or Concise. The Verbose command is a two-way switch selecting either "Long" or "Short" (another name for "Normal"). It tells you the setting it is changed to. The default is Short mode. Long mode gives additional prompting for your input.

<- Koncise

The Koncise command is a two-way switch selecting either "Concise" or "Normal" (the default Short mode). It tells you the setting it is changed to. Concise typeout mode further shortens some of the prompting for your input. It is meant for advanced users only.

<- Inclusion of length in header

This command is a two-way switch for whether headers include in the Subject field the number of characters in the message. The default is that the length not be included. However, when you read a file initially, the length of recent messages will always be included.

<- Zap profile [confirm] <RET>

The Zap profile command asks you questions allowing you to customize your future use of MAIL. Your answers are recorded, or changed, in a "user profile" (a file you manipulate only with the Zap command). Typing <CTRL-N> will exit to the top level of MAIL. Typing "e" at any point will "Exit" the dialog and ask you if you want the changes to be permanent. At any point, type "?" for a list of alternative responses. The following is a summary of the questions:

(1) Normal (Short), Long (Verbose), or Concise (Koncise) typeout mode?

(2) Always include the length of messages in all headers listings?

(3) When in SEND (for any of the Send, Forward, or Answer commands), when you type <CTRL-N>, do you want to abort Send without confirmation?

(4) Do you want to be required to confirm all commands with a single return?

(5) Do you want to be told that some messages have been "not examined" whenever you try to quit MAIL (via the Quit or Exit commands)?

(6) Do you want to automatically receive copies of your answers to messages?

(7) If the answer to the previous question is yes, then you will be asked if you want to save all your answers on the file SAVED.MESSAGES.

(8) Do you want a list of headers for all messages:

(a) being deleted with either the Move or Delete commands?
(b) being copied with the Put command?
(c) being listed with the List command?
(d) being marked or unmarked with the "'" or "-" commands?

At the end of the dialog, you will be asked if you want these changes to be permanent. If you answer "y" (for "yes"), then each time you start up MAIL, the seetings you specified will be in effect. Otherwise, they are set only for this MAIL session and are NOT permanent. In either case, they can be changed again with Zap at any time.

<- :

Typing a colon (:) displays the current date and time.

<- ' mark msgs MESSAGE-SEQUENCE

Typing an apostrophe (') marks all the specified messages to be "examined", so MAIL will treat them as if they have been typed or listed even though they may not have been seen.

<- - unmark msgs MESSAGE-SEQUENCE

Typing a dash (-) marks all the specified messages to be "not examined", so MAIL will treat them as having NOT been typed or listed, even though they may have already been seen.

<- ; COMMENT

Typing a semicolon (;) allows you to type a comment ignored by MAIL. The main use of this command is to talk with somebody via linked terminals while you are in MAIL. What you type has no effect on your mail; it is simply displayed. For "COMMENT" you may type any characters except <RET>, <CTRL-Z>, and <CTRL-N>, which return you to command level. Two other characters have special effects. <DEL> (the key labeled "DEL") displays "XXX" and is useful for indicating that.the previous word (or phrase) should be ignored. <LF> (the LINE FEED key) will continues displaying the typing on the next line so you can type more than one line of text. The standard Executive editing characters (e.g., <CTRL-A>) are treated as any other characters and perform no special function.

RECEIVING NEW MESSAGES WHILE USING MAIL

When you type a command or a command finishes execution, MAIL checks to see if your currently active message file, usually MESSAGE.TXT, has received a new message. If it has, it notifies you and prints the headers for the new messages. The current message number is not changed. MAIL then executes your command or returns to command level, accordingly.

ERRORS WHILE READING A MESSAGE FILE

When you ask for a file to be read in MAIL (either when first entering MAIL or with the Read command), the file MUST be in the "message file format". If MAIL recognizes that the file does NOT conform to this format, you will be told so. It could be:

(1) The file is not a message file.

(2) It is a message file with a bad first line (probably a blank line).

(3) It is a message file with a "hole" at the beginning.

In cases 2 and 3, first try logging out, logging back in, and calling MAIL. If the problem persists, send a message to FEEDBACK indicating the nature of the problem and any error messages that were displayed.

COMMAND SUMMARY

Command Character	Meaning
A	Answer message number: MESSAGE-NUMBER Reply to whom the message is: F {From} <RET> {same as F} T {To list plus original sender} C {Cc list plus To list plus original sender} A {All; same as C}
B	Backing up -- previous message is:
C	Current message is # of # messages in file FILENAME
D	Delete (message sequence) MESSAGE-SEQUENCE
E	Exit and update old file FILENAME [confirm] <RET>
F	Forward (message sequence) MESSAGE-SEQUENCE
G	Go to message number: NUMBER
H	Headers (message sequence) MESSAGE-SEQUENCE
I	Inclusion of length in header
K	Koncise {provides shorter prompting}
L	List (message sequence) MESSAGE-SEQUENCE on file: FILENAME
M	Move (message sequence) MESSAGE-SEQUENCE into file name: FILENAME
N	Next message is:
O	Overwrite old file FILENAME [confirm] <RET>
P	Put (message sequence) MESSAGE-SEQUENCE into file name: FILENAME
Q	Quit [confirm] <RET>
R	Read file name: FILENAME
S	Send [confirm] <RET>

T Type (message sequence) MESSAGE-SEQUENCE

U Undelete (message sequence) MESSAGE-SEQUENCE

V Verbose {provides more prompting}

W Write sorted file FILENAME [confirm] <RET> {sorted by arrival time}

Z Zap profile [confirm] <RET>

' Mark msgs MESSAGE-SEQUENCE {as Examined}

- Unmark msgs (message sequence) MESSAGE-SEQUENCE {as Not Examined}

: {prints the time and date}

; COMMENT {<RET>, <CTRL-Z>, or <CTRL-N> returns you to command level}

? type cmd. char. for its description, ? for summary ...

^ {same as Backing up}

<LF> {same as Next}

<CTRL-E> exec [confirm] <RET>

<CTRL-H> {same as Backing up}

Abort commands with <CTRL-N>.

Confirm commands with <RET>.

CREDITS

This manual is based on material from a document written by John Vittal of USC's Information Sciences Institute, and was produced from a file residing on our AUGMENT computer facilities.

EXPRES Project
Solicitation for Research Groups

The Office of Information Systems of the National Science Foundation (NSF) has initiated a project called "EXPerimental Research in Electronic Submission" (EXPRES). Its purpose is to improve the ability of the Nation's research community to exchange documents containing text, image and graphics. The vehicle for experimentation will be the NSF proposal development, submission and review process.

Awards will be made to research groups to study the problems of compound document transmission among different hardware/software environments. The project duration will be three years, beginning in September, 1986. NSF anticipates making awards to two research groups.

Proposals are invited from research groups consisting of U.S. academic or nonacademic, profit or nonprofit organizations.

Proposals must be received on or before Friday, August 1, 1986 to be considered for award under this solicitation.

BACKGROUND

The Problem:

Rapid communication of new ideas and research results is essential to progress in the scientific and engineering communities. There has been a recent explosion of acquisition and utilization of computing equipment in many aspects of research involving computation and sophisticated information processing.

This should permit exploitation of current and near-term future technology for prompt and faithful communication of scientific information in electronic form, thereby improving the efficiency and productivity of the Nation's scientific and engineering research enterprise.

Unfortunately, in the current state of diversity in computing environments, there are substantial problems in the transmission of even relatively simple documents among scientists. The major difficulties stem from the lack of suitable standards and protocols so that information can be transmitted and shared among multivendor equipment environments.

The Goal:

To move to a situation in which the scientific and engineering research community, building on computing equipment and facilities currently available, will benefit from emerging technologies and will be able to exchange compound documents among dissimilar (multivendor, multiuniversity, multidisciplinary) hardware and software environments. (Compound documents are those which may contain text with numerical and symbolic data, images, drawings, charts and graphs, and photographs.)

Meeting this goal will involve attainment of important subordinate goals:

- bringing the scientific and engineering research community and industry together to address the special needs of the community
- fostering the development and widespread adoption of technologies which can support electronic manipulation and exchange of compound documents
- facilitating migration of the research community's data communications and information processing infrastructure to standardized, interoperable systems without undue delays or expense

The Approach:

The National Science Foundation has initiated a project called "EXPerimental Research in Electronic Submission" (EXPRES). The vehicle for experimentation will be the NSF proposal development, submission and review process, which embodies all aspects of the problem (high volume, geographically dispersed community, and multivendor environments) but permits a measure of control necessary for experimentation.

The National Science Foundation will make awards to groups to perform research on overcoming the barriers to compound document transmission in the multidisciplinary, multiuniversity and multivendor environment. The research effort will include the design and installation of pilot systems with the aim of expanding them. Systems characterized as "high-function, single vendor" or "medium/limited-function, multivendor" could be used as a basis for the project.

The awardee(s) will cooperate with industry and government initiatives advocating and promoting interoperability (communication among multivendor equipment and software) based on the International Standards Organization Open System Interconnection (OSI) standards.

The National Science Foundation: NSF faces a subset of the communication problems of the Nation's research community. Scientists and engineers at about 2,400 organizations submit proposals to the National Science Foundation each year; the vast majority of these organizations is educational. NSF receives approximately ten copies of 37,000 proposals annually. Each proposal averages 50 pages and many contain more than 100 pages. Most contain not only text but images and graphics; some contain black and white or color photographs. After their initial review at NSF over half of these proposals are mailed to 6-8 reviewers for analysis. Responses from the reviewers are mailed to NSF.

Paperwork for recommended awards is forwarded through the approval and processing chain, and notification of award is mailed to principal investigators' institutions.

Research groups: This solicitation is intended for research groups expected to consist of a small number of universities and colleges and hardware and software manufacturers. Awards to two research groups are planned, although more or fewer could be made. The research groups will perform the following activities during the three year project:

1. Install and demonstrate pilot proposal submission, transmission and processing systems, with interconnected workstations at several sites including NSF.
2. Exchange compound documents successfully with NSF initially and, later, with other sites.
3. Expand the pilot project, in cooperation with other awardees, to achieve a "seamless" system capable of transmitting and processing compound documents among heterogeneous environments.
4. Participate in government and industry activities related to the formulation and practical implementation of the International Standards Organization Open Systems Interconnection architecture, in particular those aspects dealing with file transfer and management (FTAM), message handling (MHS/X.400), document architectures and document interchange formats, and related standards. Among the groups most important are the American National Standards Institute (ANSI), the National Bureau of Standards Implementors' Workshops, the TOP/MAP User Groups, the Network Users Association, and the Network and Telecommunications Task Force of EDUCOM.
5. Study the academic research community to determine the status of existing and planned computer and networking equipment and software.
6. Develop documentation for use by other institutions in implementing and using the EXPRES systems.
7. Generate the technical specifications necessary for other compound document transmission and handling systems to

interoperate with the EXPRES systems, and otherwise cooperate with additional groups in interoperating with the pilot systems.

Research areas: It is anticipated that research in the following areas will be required during the project:

- document representation and management
- networking protocols
- authentication, certification and security
- data compression
- data management
- high volume image transmission
- graphics
- human-machine interface
- user acceptance and impact on work and social patterns

Electronic proposal submission: The EXPRES experimental design will be based on the following model for electronic proposal preparation, submission and processing.

- The principal investigator prepares the proposal using a computer workstation
- The proposal is forwarded through institutional channels, to the Research Administration Office where it is approved by the authorized officials
- It is then submitted over NSFNET or another wide-area network to NSF
- A receipt acknowledgement is sent to the submitting institution
- The proposal is entered into the computerized processing system at NSF and assigned to a program manager
- After initial review, the program manager selects reviewers
- The proposal and reviewing instructions are sent to the reviewers
- Reviews are completed and sent back to NSF
- The program manager evaluates the reviews, and decides to recommend award or declination of the proposal
- For recommended awards, the necessary forms are completed and sent with the proposal through division and directorate approvals, then to the administrative divisions for final processing
- For declinations, the decision justification and proposal are sent to the division director for concurrence
- The notification of award or declination is transmitted to the institution

It is expected that paper copies of the proposal and forms could be generated at any stage in this process, but transmission and handling would be primarily of electronic media.

Cost sharing: The current total level of NSF funding available for this EXPRES activity is approximately $2 million per year. Accordingly, it will not be possible for NSF to bear all of the costs associated with the project, and significant cost sharing by the research groups may therefore be necessary. (See Evaluation of Proposals).

NSF funding will be provided for such expenses as:

- management of the research groups
- participation in the communication standards meetings and workshops sponsored by the National Bureau of Standards and by industry
- studies of the special information-processing needs and requirements of the scientific and engineering research community
- preparing, performing and evaluating the proposal submission and processing demonstrations
- disseminating the results and encouraging broader participation by the academic research community

However, NSF will not provide funds for purchase or upgrades of equipment.

Standards: The project will use commercially available hardware and software to the extent possible. Initially, the EXPRES pilot installations are expected currently to utilize and take advantage of NSFNET. Note that NSFNET conforms to DARPA protocol standards and will migrate to the international Open Systems Interconnection standards as they become available. Development of the OSI standards for data interchange, particularly those regarding document architecture and multimedia transmission, will be monitored by NSF, the research groups, an advisory panel and a contractor (the technical coordinator). They will assess the completeness of the OSI standards, and the degree to which implementation is practical or a migration path is evident. If necessary, the group will establish interim specifications for use in the EXPRES demonstrations. They also will develop criteria to be used by the technical coordinator at stages in the project to assure conformance with the agreed-upon standards. The interoperability demonstration will be directed by the contractor in conjunction with NSF, and will be the final test of adherence to the standards.

PROPOSALS

Who may submit: Proposals are invited from research groups consisting of any combination of profit, nonprofit, nonacademic or academic organizations. The research group must be headed by one of its members which will act on behalf of the others, and will serve as the legal awardee.

Proposal contents: Each proposal should reflect the proposing research group's interests and capabilities. It should discuss the features of the proposed electronic compound document handling and transmission system in sufficient detail to allow evaluation in accordance with the criteria listed later in this announcement. Proposals should be prepared in general accordance with NSF document "Grants for Scientific and Engineering Research", NSF 83-57, and should include the following:

1. NSF Cover Page (NSF 83-57 Appendix II)
2. Table of Contents with page numbers keyed to the major sections of the proposal
3. Detailed descriptions of the proposed system and management approach. This will be the main body of the proposal and should include the following:

    - a description of scope of participation in the project, including institutional and industrial participants
    - a description of the system(s) being used as a basis of the proposed approach
    - any modifications expected for conformance with the existing protocols in the academic research community and with the international OSI standards
    - research required
    - the schedule for installation of the pilot system
    - organization of the research group and division of effort among the group members
    - key personnel to be assigned to the project and the percentage of time they will be available
    - resumes of key personnel
    - experience of the research group members applicable to EXPRES

4. A proposed budget for each year of the project showing the total operating budget of the research group including NSF funds and other sources.

Mailing instructions: Fifteen copies of the proposal should be received on or before Friday, August 1, 1986, addressed to:

Data Support Services Section
Attn: EXPRES Project
National Science Foundation
Washington DC 20550

NSF 83-57: Copies of "Grants for Scientific and Engineering Research" (NSF 83-57) can be obtained without charge by contacting:

Supply Services Unit
National Science Foundation
Washington D.C. 20550

EVALUATION OF PROPOSALS

Proposals will be evaluated in several stages which may include mail and panel review, site visits, and several levels of NSF review. The panel will consist of NSF scientists and selected experts from industry and academia.

Proposal evaluation criteria are fully described in NSF 83-57. The four criteria are: (1) research performance competence, (2) intrinsic merit of the research, (3) utility or relevance of the research, and (4) effect of the research on the infrastructure of science and engineering. While all these criteria are important, the most weight will be given to (4).

Within the above-mentioned criteria, specific consideration will be given to the following items in evaluating how research groups would contribute to the objectives of the program:

- Clear demonstration of understanding of the problem, and both the short-term and longer-term objectives of NSF and the academic science and engineering research community
- Likelihood of establishment of a model interoperable environment among multidisciplinary, multiuniversity, multivendor users suitable for wider adoption at other sites
- Ability for the compound document exchange system to interoperate with the existing infrastructure, including NSFNET and the NSF distributed system network
- Competence of research group and management of group, including familiarity with technical issues of electronic mail and transmission of compound documents
- Contributions from non-NSF sources, including financial, hardware/software and human resources
- Short-term impact of the pilot projects on research proposal submissions of participating institutions
- Plan for successful demonstration of operation and interoperability in a heterogeneous environment
- Extent of impact on the scientific community and infrastructure; in particular, the breadth of institutional and industrial representation and participation in the project

- Availability of credible compound-document product capable of supporting the electronic creation and editing of text, charts and line drawings

AWARDS
NSF expects to announce the awards in September 1986.

MEETING

The National Science Foundation will conduct a meeting for all potential proposers to meet with NSF officials for the purpose of receiving clarifications and for questions to be asked and answered in a public forum. The meeting will be held at NSF, 1800 G Street NW, Washington, D.C. in the Board Room, room 540 on Friday, June 13, 1986 at 9:00 am. Please call David Staudt at (202) 357-7448 by Monday, June 9 to indicate the number of persons who will attend.

****

The National Science Foundation provides awards for research in the sciences and engineering. The awardee is wholly responsible for the conduct of such research and preparation of the results for publication. The Foundation, therefore, does not assume responsibility for such findings or their interpretation.

NSF welcomes proposals on behalf of all qualified scientists and engineers, and strongly encourages women and minorities to compete fully in the research program described in this document.

In accordance with Federal statutes and regulations and NSF policies, no person on grounds of race, color, age, sex, national origin or handicap shall be excluded from participation in, denied the benefits of, or be subject to discrimination under any program or activity receiving financial assistance from the National Science Foundation.

The Foundation has TDD (Telephonic Device for the Deaf) capability which enables individuals with hearing impairment to communicate with the Division of Personnel and Management for information relating to NSF programs, employment, or general information. This number is (202) 357-7492.

Dataletter

Exxon's latest enterprise is zoned for offices

Exxon Enterprises, the venture capital arm of the oil giant, is funding a project to develop what it has described to sources as an "integrated electronic office network." Headed by RCA's past director of corporate management information systems, John O'Neill, the project, located in Princeton, N.J., is named Xonex, which could also become the corporate name when a company is formed. Security about the details of the project are extremely tight, which has resulted in much speculation, but little detail. About all that has emerged is that the firm is designing management workstations that will integrate filing, database access, word processing, electronic mail, the telephone, and other key office functions in a distributed environment. It is also believed that the firm is working on a super-controller to connect all these elements, so that each workstation can vary depending upon need.

Store-and-forward facsimile switch to speed machines

Compression Labs Inc., a tiny specialty electronics firm from Cupertino, Calif., is causing the biggest stir in the facsimile field since Graphic Sciences announced its networking concept one-and-a-half years ago. Compression Labs recently announced a store-and-forward facsimile message switch with intermachine and teletypewriter compatibility, based upon a patented coding scheme that company spokesmen say is five times more efficient than the widely used run length coding (RLC). If that's true, the company will provide other carriers with a rapid way to catch ITT, whose software-troubled Fax-Pak network is scheduled for introduction this fall. In addition, Compression Labs says the switch will enable intrafacility facsimile networks to send or receive a page at subminute rates using souped-up 6- and 3-minute machines. Already, Compression Labs is negotiating with every international record carrier, some new start-ups, and facsimile vendors to effect these changes.

A big standards task for a tiny company

A small, Boston-based company named 3Com Corp. plans to develop a set of specifications for all-inclusive, wideband, local computer networks and then convince the highly individualistic vendors in the data communications industry to adopt it. The move is designed to short circuit the normal standards development process, which can take more than five years. 3Com is headed by Robert M. Metcalfe, the developer of the Ethernet, which was one of the early local wideband networks.

Western Digital prepared to show its X.25 chip

Western Digital Corp. has completed development of its X.25 chip and will begin offering samples of the new product to manufacturers this fall. After one-and-a-half years of engineering, research, and consultation with Telenet and Datapac engineers, the Newport Beach, Calif., firm will be the first company to offer the physical and link control levels (LAP-A only) of X.25 on a 48-pin chip. "Right now, Level 3 (network control) is too unsettled," says a Western Digital official. "When it's firmly specified by CCITT, we'll get it on the chip." Production of the Model WD2501 chip is slated for early 1980.

Viewdata service scheduled for the fall

General Telephone & Electronics, which has a license on the British Viewdata network in the U.S., reportedly is going to offer its first commercial service in September—a nationwide real estate network to aid in corporate relocations.