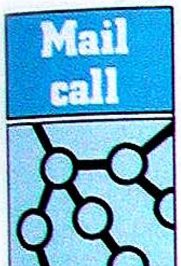
Malcolm Hamer and Jim Heilmann, Citicorp, New York, N. Y.

# How one firm created its own global electronic mail network

## A pre-X.400 solution overcame vendor incompatibilities while integrating five different software packages.




How does a company provide 19,000 users with intercommunications when it has a network of 47 processors, each running one of five different electronic mail software packages? The answer, in Citicorp's case: Give all users access to a single common directory. The Citicorp configuration is the result of a three-year plan to interoperate different vendors' electronic mail software packages. The company used a set of internal Citicorp standards, known as the Citidex Electronic Mail Standards.

The evolution of Citicorp's intracompany electronic mail service started in 1981 with a pilot installation in London. The main ingredients were a DEC (Digital Equipment Corp.) PDP-11/70 computer plus the popular off-the-shelf electronic mailbox software package, Comet. The pilot was initially limited to Citicorp's London staff, but it became so talked about that personnel at other European branches started to ask for mailboxes on Comet. The computer was connected to Citicorp's global telecommunications data network, and soon staff throughout Europe were intercommunicating via electronic mail—named Citimail.

Use of Citimail then spread to other regions, and by mid-1983 Citimail was accepted as the primary means of international person-to-person nontransactional communications within the corporation. (All communications regarding financial transactions continue to be conducted via a totally separate message-switching network, which has a high level of security and the necessary audit trails to guard against misplaced messages.)

Comet (supplied by Maxcom, based in Waltham, Mass.) is designed for the DEC PDP-11 series of machines. A PDP-11/70 or -11/84 processor, running Comet, can support about 64 concurrent sessions. This is the typical session demand created by 1,600 users in one time zone, or as many as 4,800 users spread around the globe (so that advantage can be taken of nonoverlapping working hours).

Comet's efficiency results from a conservative database structure. But the database is not continuously self-reorganizing, so the application has to be "taken down" (preventing users from gaining access) for a few hours on weekends to run database reorganization routines. Also, Comet has only a primitive line editor, rather than the kind of full-screen editor available on more recent packages such as DEC's All-in-One. (An example of the Comet line-editor's crudeness: The operator cannot back up the cursor beyond the line on which it is.)

### Easy to manage

Nevertheless, Comet seems to gain rapid acceptance by new users, possibly because of its very simple command structure: There is only one command level, no menus, and only a handful of command verbs to remember.

In 1983 another PDP-11/70 had to be added in London. (By this time a backup machine had already been added to the original configuration to guarantee continuity of service in the event of a hardware failure.) Demand for Citimail grew so rapidly that this second node was full within a few months, so that a third node was needed. This third node was placed in New York to serve Citicorp staff in North and South America. It consisted of a PDP-11/70 plus a backup. The multiple-machine environment, with nodes in London and New York, was handled using the standard multinode Comet software. This software makes node intercommunications transparent to the user.

In parallel with the growing popularity of Citimail, several areas of the corporation had started to look for integrated office automation (OA) and electronic mail solutions to their internal day-to-day business requirements. Three organizational areas implemented such packages. One area de-

## Message types and type codes

| USER MESSAGES | |
|---|---|
| **CODE** | **TYPE** |
| 01 | ORDINARY TEXT MESSAGE |
| 02 | USER-REQUESTED ACKNOWLEDGMENT |
| 03 | SYSTEM-REQUESTED ACKNOWLEDGMENT |
| 05 | NEGATIVE ACKNOWLEDGMENT (MESSAGE UNDELIVERABLE) |
| **DIRECTORY UPDATE MESSAGES** | |
| 50 | ADD A NEW LOCATION CODE TO THE DATABASE |
| 51 | MODIFY AN OLD LOCATION CODE IN THE DATABASE |
| 60 | ADD A NEW USER |
| 61 | DELETE AN OLD USER |
| 70 | ADD A NEW DISTRIBUTION LIST |
| 71 | DELETE AN OLD DISTRIBUTION LIST |
| 80 | ADD MEMBER(S) TO A DISTRIBUTION LIST |
| 81 | DELETE MEMBER(S) FROM A DISTRIBUTION LIST |
| 90 | CHANGE A USER'S NAME OR A DISTRIBUTION LIST NAME |
| 91 | CHANGE A USER'S NODE NUMBER |

ployed a network from San Antonio, Tex.-based Datapoint, whose electronic mail feature is called E-Mail. This feature offered close integration of electronic mail with word processing, plus full-screen editing of electronic mail messages. Two other areas chose DEC's All-in-One package, which has similar characteristics.

By late 1983 it became clear that a problem was arising. Users on the local OA-based packages wanted to exchange messages with Citimail users, who numbered about 6,000. The only way they could do this was to be enrolled on both services, using the OA-based service for within-division communications and Citimail for communications with other areas, particularly overseas branches. To try to solve this problem, a committee was set up to investigate how the various packages could be made to communicate with each other.

While less ambitious than the CCITT X.400 standards, the internal Citicorp standards are dominated by a concern with directory and addressing issues. These issues have a very different scope with an intracompany network than with public electronic mail networks.

### Arguing the philosophy

The investigating committee had one major advantage over the CCITT X.400 committees: The eventual population that would be using the consolidated network was limited to a well-defined set of human beings: Citicorp employees, who currently number about 100,000. This makes the addressing problem many orders of magnitude simpler than the public electronic mail problem of potentially addressing any human being on the planet. Even so, some members of the committee argued that it would be impossible to continue with the basic Comet scheme of addressing people by their names. They proposed, instead, the use of personnel numbers.

After some lively debates, the personnel-number approach was rejected because it was awkward and redundant. In any case, users must supply enough information about a desired addressee to uniquely identify that addressee in the directory. So why not use that information directly? In the end it was agreed to construct a unique identifier for every employee by adding a suffix to each name. The suffix took the form of *Location:Group*, where Location identifies the geographical location, and Group identifies the person's functional working area. In the case of common last names, middle initials would be pressed into service to provide a unique key if first name, location, and group all coincided.

It was also agreed that the location code would be based on IATA (International Air Transport Association) airport codes, to minimize the number of characters appearing in message headers. (For example, Los Angeles would be identified as LAX.) Because some of the airport codes are somewhat obscure, it was decided to prefix them by a two-letter region code, to guide users to where the city might be. So the location code for Los Angeles becomes USLAX. Brussels takes the region code EU for Europe, so the location code for Brussels is EUBRU. Hong Kong is APHKG, where AP stands for Asia/Pacific.

The group codes were based on abbreviations already widely used in the Citicorp telephone directory, such as INV for Investment Banking Sector, IND for Individual Banking Sector, IB for Institutional Banking Sector, and AUD for Audit Division. So, three similar entries in the directory might be: John A. Smith (USLAX:INV); John B. Smith (EUBRU:IND); John B. Smith (APHKG:IB).

Another point to which the committee members agreed: Users would not, generally, key in these location/group suffixes, such as "(USLAX:INV)"; in most cases the suffix would be appended by the electronic mail software package. If, in the *To:* line of a message, the user keys "John A. Smith," or even just "J. A. Smith" where there are no other J. A. Smiths, then this will be accepted immediately and confirmed by the electronic mail software.

### 'Will the real John Smith . . .'

The addressee confirmation takes the form of a repetition of the *To:* or *CC:* line after the one entered by the user. It may differ from what the user enters: Names are shown as they appear in the directory ("John A. Smith" where the user typed just "J. A. Smith"). Also, the suffix is appended. The confirmation reassures the user that the correct John A. Smith has been identified. However, if the user types an ambiguous name, the electronic mail package will offer a choice of possible names, giving the suffix for each.

For example, the dialogue might go as follows: (The regular text is that typed by the user; the italics represents that generated by the electronic mail software package.)

*To:* John B. Smith
*Ambiguous name.*

*Do you wish to see 2 similar names (Y/N)?* Y
*1 John B. Smith (EUBRU:IND)*
*2 John B. Smith (APHKG:IB)*
*Corrected name or number:* 2
*To: John B. Smith (APHKG:IB)*

In this example, the final line of text is the addressee confirmation for the *To:* line.

Users who are aware of the ambiguity (from a previous attempt to send a message to one of the John B. Smiths) can key the name and suffix themselves, to avoid being taken through the above process. Or they may type just the name and answer the resulting prompts. Experience has shown that only a low percentage of names is ambiguous without the suffix, so that it is rare for users to have to go through a dialogue like the one above.

What turns out to be much more common is that the user is unsure of how the first name or initials appear in the directory and will therefore type just the last name. This results in the electronic mail software package offering a number of possible names, from which the user makes a choice. Users in these circumstances have reported (in various surveys and interviews) that they have found the suffix extremely useful in identifying the user they are seeking. They are more likely to know the city and the part of the company in which the user works than the first name and initials. Users also report that the suffix has been useful in identifying where people are located. (On some messages with many addressees, it was previously difficult to sort out who one's fellow addressees were.)

Other important philosophical points that the committee soon accepted were:

- Every electronic mail node in the corporation, once linked with the other nodes, must hold a directory of all users on all nodes. Users then have the perception of all being on the same node, since they can then send messages to anyone on any node without first identifying the addressee's node.
- The complete directories held in each node must be kept fully synchronized (identical) using minimal manual operations. Human intervention for the directory-update function should be performed only by the System Administrator of the node that has the mailbox of a user whose directory entry is being created or changed.
- No node is allowed to send a message to another node without every addressee being positively identified from the complete directory. So, barring directory synchronization failures, internodal messages should always be accepted at the receiving node, and manual intervention to deal with undeliverable messages should be a very rare occurrence. This is in contrast to many distributed public networks, where "wild" messages (those with unvalidated addresses) can be sent to another node.

## Seamless and hands-off

The investigating committee's next assignment was to define a set of internodal message types. These message types would be needed to achieve the committee's dual objectives: a "seamless" integration of the different vendors' packages and a minimal human intervention in the maintenance of directory synchronization.

Two general classes of messages were specified: user and directory update. User types were subdivided into ordinary text and acknowledgments.

The vast majority of messages passing between nodes during normal operation are ordinary text. A small proportion are acknowledgments. The latter generally occur only when the sender specifically requests that receipt of the sent message be acknowledged.

To allow for the possibility of a node operating in a positive-acknowledgment mode for all messages, two types of acknowledgments were defined: sender-requested and network-requested. Sender-requested causes the destination node to issue an alert that receipt of the message will be acknowledged if it is read.

Network-requested, on the other hand, is generated automatically by the receiving node when the associated message is placed in the user's in-box. It is thus proof that it got to the destination node but does not mean that it has been seen by the addressee. If necessary, a message may be flagged for both types of acknowledgment.

The other type of message—directory update—represents a very small proportion of traffic. It is mixed in with the user messages on the same physical internodal links. At most, there are a few hundred updates a week across the whole network, which represents less than one percent of the total internodal traffic.

Before looking at the various types of directory update messages that the committee defined, it is necessary to understand the directory-database structure being considered. This structure represents the minimum requirements for an electronic mail package that is to meet the Citidex standards. (Individual electronic mail packages may have additional elements in their directories.)

The model database structure consists of three parts: an individual user part, a distribution list part, and a location-code table. A distribution list is thought of by its name, such as Electronic Mail Committee, plus a set of "pointers" that point to the directory entries of individual users who are members of that list. This means that the directory entry for users may be changed without having to make any changes to the distribution lists on which they are.

## Character count

The model for the individual-user part of the directory database consists of:

- Name (32 characters maximum)
- Location code (8 characters maximum)
- Function code (8 characters maximum)
- Node number of user's mailbox (4 digits)
- Last update sequence number (2 digits)

The committee allowed for eight characters in the location code, even though the initial scheme used only five. The use of four digits for the node numbers was meant to reduce to nil the possibility of running out of node numbers. In practice, it is unlikely that the number of nodes would exceed 50.

The model for the distribution list part of the directory database consists of:

- List name (32 characters maximum)
- Location code (8 characters maximum)
- Function code (8 characters maximum)
- Last update sequence number (2 digits)
- Pointers to individual directory entries of list members

All location codes must be explicitly defined in the database's location-code table. All new entries or changed entries in the individual-user part of the database must be checked against this table—to make sure that the location code is a recognized one—before the directory update is allowed. A consequence of this procedure is that, before a node can advise the other nodes of the first user in a new location, the location code for that location must be broadcast to the other nodes. Only then can users with that location code be added. This structure was defined in order to eliminate the possibility of "lost" users (without a recognized location) being entered in the directory due to keying errors by System Administrators.

To keep directories synchronized, 10 types of directory update messages were defined (see table). These types were needed to cover all the possible update actions relating to individual users' directory entries, to distribution lists, and to the location-code table.

The assumed creation method for distribution lists is to first create an "empty" list (message type 70) and then add members to it (message type 80). This was thought to be a cleaner method of handling list creation than to include the initial members in the list-creation message.

## Primitive is better

The next step was to define the exact format of the messages that would be passed between the electronic mail nodes. While the idea of a layered protocol like X.400's had intellectual appeal, it was agreed that a more primitive arrangement would be easier to implement in the short term. The approach selected was to use the traditional message switching approach of a message header with well-defined fields and field labels, plus a unique message-termination character.

Figure 1A is the format of the message as it passes between two nodes. At the destination node, the message is presented to the recipient in whatever format that user's package employs. An example of such a format is shown in Figure 1B.

In the internodal message (Fig. 1A), the message header consists of a number of fields, each labeled with a three-digit field label preceded and followed by a colon (for example, :801:). Each field is terminated by a "carriage return."

Only the second and third digits of the field labels are significant. The extra digit, which is always 8, was added so that, in the future, nonmail message types—identified by a different initial digit—could be intermixed with mail messages on the same network.

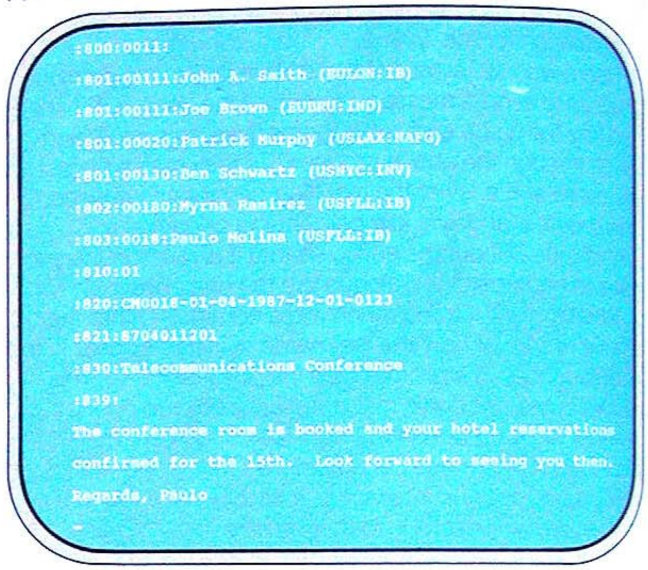
The message terminator is a single hyphen on a new line, followed immediately by a carriage return. The internodal message-handling software prevents—by "space stuffing"—a hyphen in the message text from causing a premature termination of a message. Space stuffing inserts a space after a text-resident hyphen. A hyphen is only a terminator if it is in the left-most column and has no character or space between it and the carriage return.

## Easier debugging

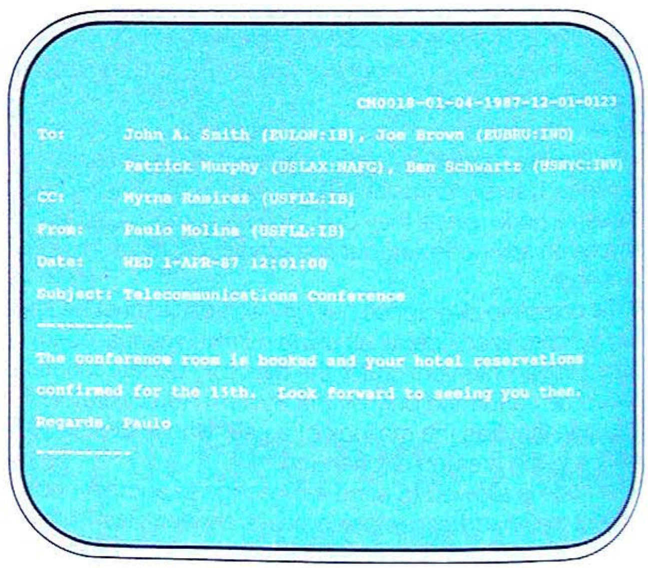
This method of message termination was chosen over a nonprintable reserved character because the committee had a strong preference for using recognizable printable characters wherever possible. It was felt that such a method would make software debugging easier: sample messages could be readily analyzed from a printout, instead of an operator having to interpret strange charac-

***1. Internodal.*** *The format of the message as it passes between two nodes is shown in A. At the destination node, the message is presented to the recipient, as in B.*

**(A) EXAMPLE OF AN INTERNODAL TEXT MESSAGE**




**(B) HOW THE ABOVE MESSAGE WOULD APPEAR TO ONE OF THE ADDRESSEES**



ters displayed on a network-analyzer screen.

The committee defined 26 field labels. An ordinary text message may use up to 10 of these. Directory update messages tend to use somewhat fewer field types. In the example (Fig. 1A), field :810: contains the message type code, which determines what header fields must be present and what optional fields may be present. Field label :839: appears only in an ordinary text message and is the start-of-text indicator.

The :800: field contains the number of the node to which this particular copy of the message is being sent (node 0011). In this example, the source node will send similar copies to nodes 0002 and 0013, because the message is addressed to users on these nodes. The :801: fields contain *To:* addressees; the :802:, *CC:* addressees. The *From:* line appears in field :803:. Note that the message comes from a user on node 0018. Field :820: contains the message ID; :821:, the date/time stamp; and :830:, the subject.

The fifth digit on the end of the node number in the :801: and :802: fields is an acknowledgment-request code. 0 is for no acknowledgment; 1, sender-requested acknowledgment; 2, network-requested acknowledgment; and 3, both types of acknowledgment.

Although not illustrated in this example, fields :801: and :802: may contain distribution-list names. In this case, the node number that precedes the distribution list name is set to a "wild card" number of 0000, because distribution lists may contain members on more than one node. The receiving node must disassemble each list to determine if any of the list's members are on that node.

If the distribution-list name is the only addressee of the message, the software sends copies of the message only to nodes with addressed members. (This approach prevents the sending node from aimlessly sending copies of user messages to all nodes, regardless of whether the messages contain addressees for those nodes or not.) However, if the message is also addressed to other individual users or distribution lists, disassembly of a distribution list at the receiving node may result in no users being found belonging to that node. (Disassembly of the list refers to an electronic mail software operation to place copies of the message in their proper mailboxes.) In the *To:* and *CC:* lines, as displayed on the screen of the message recipients, only the distribution-list names appear, not the names of the list's members.

## Change administration

Although user messages are intended only for the nodes of their addressees, directory-update messages are broadcast to all nodes. A directory update occurs when a node's System Administrator adds, changes, or deletes a directory entry. The update is immediately implemented on the node in question, which then automatically sends update messages to all the other nodes. A node receiving an update message automatically carries out the update, normally without human intervention. However, if a problem arises (such as an update sequence-number gap or a failure to find the original directory entry), incoming update messages will be "spilled" to the receiving node's System Administrator.

***2. Modification message.*** *This directory update illustrates a change in the user's organizational group (from IB to INV) and location (from London to Tokyo).*

**EXAMPLE OF A DIRECTORY UPDATE MESSAGE**

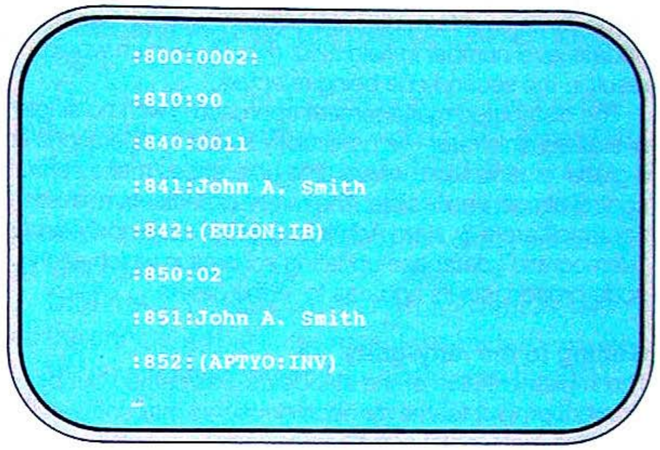



Figure 2 illustrates a modify-name and/or -suffix update. In this example, the user's name is unchanged, but he has moved to a different organizational group (from IB to INV) and to a different location (from London to Tokyo). Field :840: contains the sending-node number. This is the node on which the sending user has his mailbox. Field :850: contains an update sequence number. The directory entry at each node should have an update level of 01 prior to processing this update, and an update level of 02 after. The :851: and :852: fields contain the new directory entry.

The committee defined the formats of each message type: which fields were required and which optional; also, which could appear only once (like the *From:* field), and which could appear one or more times (like a *To:* or *CC:* field). Thus, there are four classifications for a header field: RS (required, single), RM (required, multiple), OS (optional, single), and OM (optional, multiple).

## Operational restrictions

In defining how the network would actually work, the committee established a number of rules. Some are:

- Each node or group of nodes must have a System Administrator to establish new mailboxes, to change directory entries for the node's users and for distribution lists, and to remove users who leave the company.
- No "proxy" additions would be allowed. That is, one node's administrator cannot force another node to create a new mailbox for a user on that other node. The rationale: 1. None of the software packages was capable of opening a new mailbox on the basis of an incoming update message; 2. There are other pieces of information required to open a new mailbox, such as the new user's initial sign-on password. Mailbox-creation is thus under the sole control of the System Administrator of the node in question.
- In principle, a distribution list can be created by any System Administrator; also, any System Administrator can

add members to an existing list. However, the committee initially adopted a conservative operational approach to have each list "owned" by a specific administrator, who would make all the additions to and deletions from the list. This avoided having two add-member updates generated at the same time from different nodes—with the same update-level number in field :850: (Fig. 2)—which would result in the second one being rejected.

The distributed-management approach—with no single overall administrator—is particularly suited to operations in a global organization, where time differences make central control almost impossible. The major ingredients of this approach are a System Administrator for each node, and a "democratic" database-updating scheme in which each node broadcasts its updates to all the others.

**Getting to the nitty-gritty**

The implementation of the standards was carried out by different project teams for each electronic mail software package. For Comet and All-in-One, joint projects were set up with the vendors. The latter provided programmers to write add-on software modules to implement the standards.

In the case of the Datapoint E-Mail subnetwork, a team of Citicorp programmers wrote a gateway program to run on a Tandem Non-Stop processor. The gateway provided an interface between the entire Datapoint subnetwork and all the other nodes in the network. Datapoint provided no support for this project. Tandem hardware was used because the part of Citicorp that uses E-Mail had a number of other Tandem-based applications under development. The plan was to run the gateway on a machine shared with some of these applications.

The phasing of the implementation was arranged to minimize programmer time spent on testing for logical flaws. First, the standards were installed on the Comet nodes, taking over from the vendor's own multinode Comet arrangement. However, the nodes could be rolled back to multinode Comet operation if things went wrong.

The use of the *Location:Group* suffix was then introduced into the directories. This caused some initial performance problems because of the way the directory look-up algorithms worked. However, after some "fine-tuning" of the search algorithm, the average look-up time (the time that elapses from the user typing the *To:* line, followed by carriage return, to the *To:* line being confirmed back to the user) was restored to about one second per name.

Exchange of user messages between the nodes under the standards was relatively trouble-free from day one. Only the directory update messages caused any significant trouble. Initial reject rates on incoming update messages (requiring intervention by the administrator) were somewhat higher than expected. However, after further software debugging, they were brought down to a level—a few per week—that could be easily handled by administrators as a standard operating procedure.

All the Comet nodes were DEC machines, and the lower protocol layers were well taken care of by Decnet. Therefore, at the link level, the exchange of messages between Comet nodes was accomplished relatively easily.

In the second phase of implementation, work was completed on the Datapoint E-Mail gateway, and tests were started between it and the Comet subnetwork. This was somewhat more complex than the first phase because software for the lower protocol layers had to be debugged at the same time as the electronic mail data-exchange software. Initially, a proprietary Citicorp protocol was used to provide Layer 2 (Data Link) and Layer 4 (Transport) services. However, this was replaced at a later stage by X.25, combined with a proprietary Layer 4 protocol for message-delivery assurance.

**Tough task**

The most rigorous standards-testing period occurred with the message exchanges between the Comet nodes and the E-Mail gateway. The two teams had made different assumptions about what was meant in the standards. For example, clarification was needed of which fields were optional and which mandatory. In the end, the "right" answer to each difference of interpretation was decided by negotiation between the teams. As a result of this testing period, the standards were updated to make them more explicit on the points where problems had arisen.

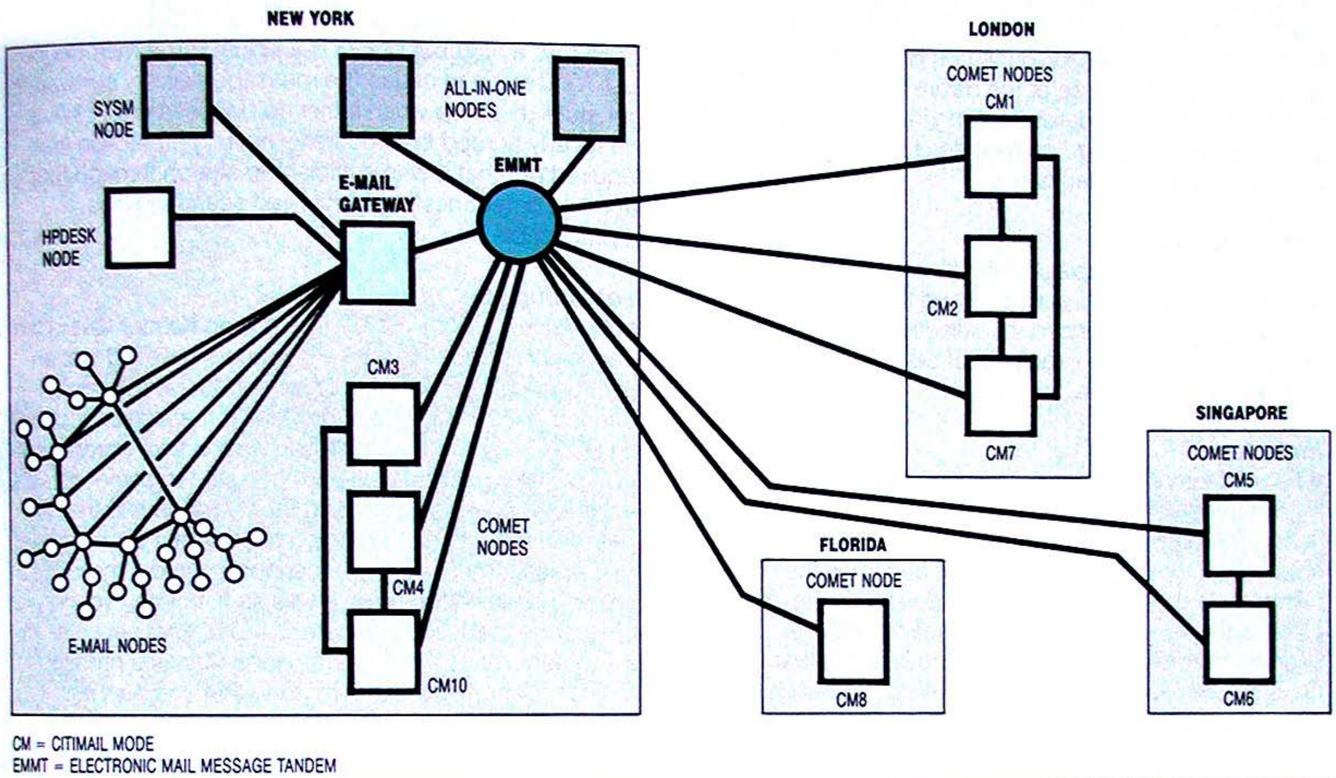
In the third phase of implementation, work was completed on adapting the All-in-One software to the standards. By this stage, the standards were well-defined and substantially free of logical flaws and omissions. As a result, the testing evinced fewer surprises than in the E-Mail/Comet tests. All-in-One was ready for interconnection after several weeks of pilot operation.

At about this time the physical interlinking of the nodes was starting to become a problem. Provision had been made in the standards for any node to act as a tandem switching point for messages between other nodes. However, this feature had not been applied in practice because too many of the nodes were already stretched to the performance limits of their hardware.

Each node was therefore connected to each other node. This was relatively easy in the case of the DEC-based nodes because of the use of Decnet. However, the connection of the E-Mail gateway to all the Comet nodes was a nightmare, since there were at that time seven separate channels (two of them international ones) to be continuously monitored. To simplify the network, an Electronic Mail Message Tandem (EMMT) was developed to allow the network to be changed from a fully connected mesh to a quasi-star configuration.

The EMMT is a DEC PDP-11/70-based application that communicates with DEC-based nodes via Decnet and with non-DEC nodes using one of several protocols that it supports. To determine the routing for a particular message—if Decnet addressing does not take the message to its final destination—the EMMT looks at the first line of the message header (the :800: field). For example, the Comet node in Singapore can deliver a message to the E-Mail gateway by using Decnet to get the message as far as the EMMT. The EMMT then accepts the entire message from Decnet, stores it on disk in case of problems, looks at the

*3.* ***The result.*** *The final network includes nine Comet nodes, two All-in-One nodes, and the E-Mail, HPDesk, and SYSM nodes—a configuration with five different software packages interworking under the Citidex standards. CM9, the 10th Citimail node (not shown), was allocated for South America but is not yet implemented.*




first line of the message header to identify the destination node, and selects the appropriate outgoing route. Then the EMMT sends the message to the E-Mail gateway, using the protocol chosen for that route.

## Continuous operation

Each non-DEC node need only be connected to the EMMT in order to achieve full intercommunication with all the DEC-based nodes on the network. In effect, the EMMT acts as the hub of the global Decnet. In addition, having no electronic mail package to support, the EMMT is a reliable Decnet node. That is, there is no risk of having its operation as a tandem point on the Decnet being disrupted by housekeeping routines that are common on the actual electronic mail nodes.

Once the EMMT was operational, the final cutover of the first All-in-One node was completed. At this point, there were three different vendors' electronic mail software packages providing a seamless, global service to about 16,000 users.

Subsequent to the initial three implementations, the Citidex standards were implemented on the Hewlett-Packard HPDesk package, which runs on an HP3000, and the SYSM package (from Boise, Idaho-based H & W Computer Systems), which runs on an IBM mainframe. Because the numbers of users on these nodes were relatively small, and because organizationally they came under the same area that uses the E-Mail subnetwork, they were connected into the E-Mail Gateway, rather than directly to the EMMT. (In principle, they could have been connected to the EMMT.)

For the System Administrators, the only significant difference between the original environment of separate subnetworks and the new integrated network is the need to handle a number of spilled messages each day. As mentioned earlier, some of these are update messages that, for one reason or another, could not be processed at the receiving node. Administrators would typically telephone each other to expedite handling a spilled update message.

The rest of the spilled messages are user types that cannot be delivered because of database conflicts. The most common cause of an incoming user message being spilled to the administrator is the user message overtaking an update message that affects a user message's addressee. The administrator finds out why the message was not processed automatically, tries to fix it, then either sends it on its way or rejects it back to the originator.

Each electronic mail package handles the generation of outgoing update messages in a different way, and it is possible for updates to lag behind user messages in some of the packages. The same result can occur when incoming update messages are placed in a queue for processing that is longer than the queue for incoming user messages.

For the user—apart from the implementation of the new addressing scheme using the location/group suffix—the operation of each node has hardly changed.

In defining the various internodal message formats, the investigating committee had to arrive at a definition of a standard set of user services. These services—in X.400 terms, user-agent services—would be available to the entire population of users in the corporate network. This was necessary because of the desire to limit the complexity of the Citidex standards, bearing in mind that they would be implemented by add-on modules for off-the-shelf electronic mail software packages.

**Software clout**

Comet, being the simplest of the three original packages (Comet, E-Mail, and All-in-One)—and the one with the most users—strongly influenced the selection of the standard set of services. Some examples of services that were and were not included:

***Included***

■ *Forwarding of messages.* Messages can be forwarded, by either the sender or recipient, to one or more other users. The original message, complete with header as it appears on the workstation screen, becomes the new message text. As an internodal Citidex message, the forwarded message has only one header with field labels such as :801:. The consequence of this rule is that users on, say, Comet, who receive a forwarded message from a user on E-Mail, will see the original message with the *To:*, *CC:*, *From:*, and *Subject* arranged in the E-Mail format—not the Comet format. This did not lead to any complaints or misunderstandings.

■ *Answering received messages.* The provision of this capability in a given software package has minimal impact on the standards, and its omission on any node does not affect other nodes. However, allowance was made for answering of messages by way of a header field, :831:, which is an undefined extra-header line. Its principal use is to enable the inclusion of the *Subject* line from the original message to which the answer is being given. Where :831: is used in this way, the committee recommended that *Re:* be inserted before the contents of the original subject line. To preserve the generality of field :831:, this *Re:* is inserted as textual content by the sending node.

■ *Distribution lists.* Distribution-list names appear in the :801: and :802: header fields. The disassembly of a distribution list into its members takes place within the software of the node receiving the message and is invisible to the user. In other words, disassembly is aimed only at deciding into which mailboxes to place the message. The distribution-list name is what users see in the *To:* or *CC:* line. Of course, a node may hold additional local distribution lists that the other nodes do not know about, provided that messages leaving the node have the actual users' names substituted for the local distribution-list name in the message header.

■ *Nesting of distribution lists.* The committee agreed that, since all software packages would be able to cope with nesting, this would be allowed under the standards. In other words, some or all of the members of one distribution list could be the names of other distribution lists. Interestingly, at one committee meeting, someone raised the question, "Do we have to check for recursive distribution-list definitions?" For instance, should the software identify and reject an attempt to add List A as a member of List C if List C is a member of List B and List B is a member of List A? This could cause the message-delivery software to get stuck in a loop when it tries to disassemble List A. It was finally agreed that recursive-nesting protection was required but that it was sufficient to search through four layers to detect nesting if unlimited searching was impracticable.

***Not included***

■ *Blind carbon copy (BCC).* There is no header field in the standards for BCC. Comet's designers believed that an electronic mail service should be conducive to good manners. Accordingly they dismissed the idea of supporting BCC. They argued that if you want a third party to see a copy of a message that you have sent to someone else, you should use the forwarding facility for getting the message to that party after you have sent it in the normal way. At least on Comet, you cannot forward a message that has been edited since it was sent, nor can you forward a message that you have composed but not yet sent. This is another example of Comet's good-manners philosophy: You cannot pretend to a third party that you sent a message that you did not really send.

■ *Silent acknowledgment of message receipt.* The issue of acknowledgments was mentioned earlier. Comet warns: "Reading this message will send an acknowledgment. Do you wish to continue?" Comet's designers considered it bad manners to let the sender know that the recipient has read the message without the recipient being aware of what is going on. Although some software packages allow the sender to have a "silent" acknowledgment, the committee specified the Comet treatment.

The inclusion of a network-requested acknowledgment was a concession to software packages with full tracking of messages. It allowed the display of "Message placed in in-box at destination node" by the sending node. The committee left it to the network implementers to decide whether they would design their software to send out all messages tagged for network-requested and sender-requested acknowledgments. In the end, nobody opted for creation of acknowledgments by default. It was generally recognized that users are annoyed by nonvital acknowledgment-requested messages.

■ *Non-text attachments to messages.* Although some electronic mail packages support the attachment to messages of nontext files (such as spreadsheets and graphics), the committee decided not to support this under the standards. Message text was defined as strings of ASCII printables plus carriage return or carriage return/line feed. This "lowest common denominator" of electronic mail service could be supported by all the software packages. The software modules written to implement the standards had to remove any attachments from messages sent outside the node (they are acceptable within the node) and interworking under the Citidex standards is now five (DEC's All-in-One, Comet, Datapoint's E-Mail, HPDesk, and H & W's SYSM). Current volume is about 200,000 messages a day (counting a message with, say, three addressees as three separate messages). Of these, about 10,000 messages pass between nodes in different locations using the Citidex standards, and about 40,000 additional pass between collocated nodes. Wherever possible, members of the same organization have been given mailboxes on the same node, to minimize internodal traffic.

While most nodes are linked via Decnet for host-to-host communications, user-workstation access is asynchronous ASCII at 300 bit/s, 1.2 kbit/s, or 2.4 kbit/s (depending on what is supported at the user's location). This access is achieved via Citicorp's Global Telecommunications Network (GTN), which has nodes (mainly CASE Communications DCX switching statistical multiplexers) in 75 countries (see "The global story"). Workstation access is always to the node on which users have their mailboxes, regardless of the location from which they are calling.

In accessing a Comet node, users operate in teletype mode and employ the simple line editor mentioned earlier. All-in-One offers full-screen editing, provided that the user has a DEC terminal—or has a PC communications package that emulates a DEC terminal. (All-in-One defaults to teletype mode—with its simple line editor—if the workstation does not respond to the DEC handshake.)

At any one time, there may be as many as 64 users accessing each Comet node. The Comet subnetwork has a theoretical capacity of close to 600 concurrent sessions. However, this level of activity does not actually occur, because of time differences.

While somewhat less elegant than the X.400 series of standards, the Citidex Electronic Mail Standards represent a significant practical step toward linking different vendors' electronic mail software packages. What is perhaps most important about these standards is that they take care of not only the exchange of user messages and acknowledgments but also the much more complex directory issue.

The committee members believe that solving the directory problem is crucial to the success of electronic mail. They doubt that Citicorp's electronic mail network would have achieved its popularity if users had been burdened with complex addressing schemes. Even something as apparently harmless as having to prefix a person's name with a node number (such as *73:John A. Smith*) would probably have been a deterrent. The way the different software packages interwork under Citidex is genuinely seamless: All users appear to be on one large node. ■

*Malcolm Hamer and Jim Heilmann are vice presidents of Citicorp. Hamer is responsible for corporate-level standards-setting in data communications. A physics graduate of Oxford University, England, he has worked in telecommunications for 17 years. He holds an M. B. A. from New York University and has co-authored three books on telecommunications. Heilmann is responsible for global Citimail development and operational standards. A Business Administration graduate of Marist College, New York, he entered the field of telecommunications 13 years ago.*