Among Debuts at Show

Network Facility Has DP, WP, Electronic Mail

By Jeffry Beeler
CW West Coast Bureau

CHICAGO — A networking facility that reportedly combines DP, electronic mail and word processing (WP) highlighted the parade of system- and terminal-related products introduced last week at the seventh annual Interface data communications conference and exposition.

Aimed at multisite businesses whose annual revenues range from $500,000 to $25 million, the Basic/Four Corp. offering is said to embody two approaches to distributed processing — a business information network and host communications support.

In the business information network, two or more Basic/Four Model 200 or 410 processors serve as stand-alone nodes built around a Basic/Four Model 730 host processor, a company spokesman explained.

The business information network also incorporates an electronic mail system which in turn consists of two modules, a "quick message" facility and a WP capability. The quick message facility permits memos to be entered at a remote CRT terminal and transmitted later during routine polling, while the word processing feature lets users generate, send and receive correspondence.

In the host communication support approach, the three Basic/Four minis use IBM 2780 and 3780 protocols to communicate bisynchronously with foreign host processors at 2,400 bit/-sec.

Both approaches reportedly include a forms entry system that allows remote

used last week's show as an occasion to announce several product enhancements, including the addition of a foreground and background partition to its 21/20 data entry and 21/40 transaction procesing systems.

Memory Price Cut

At the same show, MDS also cut by 50% the price of 16K- and 32K-byte add-on memory modules for the 21/20 and 21/40 as well as for the 21/50 multiapplication processing system.

With the price cuts, each 16K- or 32K-byte memory increment to a basic 48K-byte 21/20 or 21/40 will cost $525 and $1,175, respectively. Previously, a 16K-byte addition cost $1,100 compared with $2,350 for a 32K-byte increment, an MDS spokesman explained.

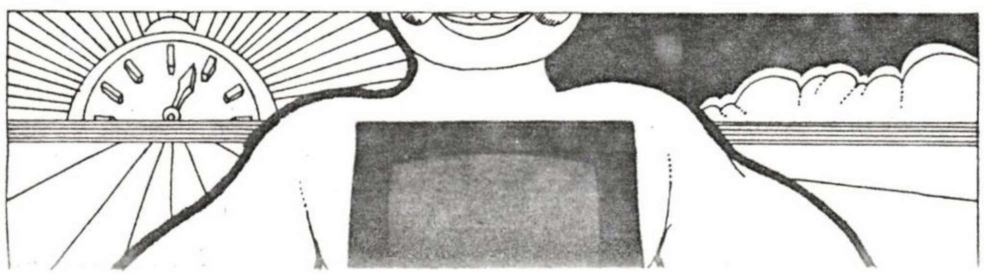
The same price cuts also apply to each 16K- or 32K-byte memory enhancement to a basic 96K-byte 21/50, he added.

CW at Interface '79

Elsewhere at Interface '79, Teleray, Inc. announced the Model 10 desktop CRT terminal and a repackaged version of its Digital Equipment Corp. UT-52 terminal replacement.

Although functionally equivalent to its predecessors, Teleray's latest UT-52 replacement has undergone some minor cosmetic changes that allow it to harmonize with typical business office decor. At $1,240, the latest UT-52 substitute costs $100 more than its predecessor, a Teleray source noted.

The Model 10 provides five field modifiers and 32 programmable functions with 16 programmable tabs and six transmission levels. Suited primarily for OEMs, the terminal costs $1,290 and will become available in May from Teleray at Box 24064, Minneapolis, Minn. 55424.



In the host communication support approach, the three Basic/Four minis use IBM 2780 and 3780 protocols to communicate bisynchronously with foreign host processors at 2,400 bit/-sec.

Both approaches reportedly include a forms entry system that allows remote users to send source data to a host site through an automatic information transfer facility and then generate a library of business forms.

Users can buy the networking facility, which consists of existing Basic/-Four hardware and is controlled by previously unavailable software, as a complete system, or they can acquire the facility's software packages separately and add modules to systems already in the field.

As a complete system, the networking facility ranges in price from $25,000 to $150,000, depending on the configuration. As separate modules, the electronic mail capability costs $1,500; the forms entry system, $1,500; and the WP feature, $12,500. They will be available in July from Basic/Four at Box C-11921, Santa Ana, Calif. 92711.

Minimate Announced

In other Interface '79 product announcements, Western Telematic, Inc. introduced a minifloppy disk-based external storage unit that reportedly attaches to smart CRT terminals and replaces paper tape, cassette or magnetic card systems.

Minimate stores more than 71,000 characters with a capacity of 560 addressable records and transmits at 110- to 9,600 bit/sec, according to a Western Telematic spokesman.

Designed primarily for store-and-forward applications, the RS232-compatible unit costs $1,295 and will become available in June from Western Telematic at 2435 S. Anne St., Santa Ana, Calif. 92704.

Mohawk Data Sciences, Inc. (MDS)

future, but developing workable systems and getting personnel to use them properly presents a stumbling block. That is the intermediate result of a multiclient-sponsored study made by Battelle Laboratories here into the future of electronic mail.

"It's a slow process," according to Richard J. Bengston, a researcher at Battelle. "In the case of electronic mail, probably nothing very important will be in use in the next five years.

"By 10 years from now, we expect there will be some pretty darn good systems operating, internally at least. Whether they will be operating among companies, I don't know."

The biggest developmental problem facing corporations today is a lack of compatibility among communications products.

"In order to take advantage of electronic mail or other electronic office stuff, you're going to have to go to a common system for the whole company, rather than letting the finance people use one thing and somebody else use something else," Bengston said in a recent interview.

"In the past there's been a functional outlook, whereby each department went its merry way."

Companies that have a hodgepodge of CPUs and communications equipment should worry, Bengston said. "Some of them are afraid to do anything about electronic mail for this very problem. Next year something else comes out that doesn't fit what we have. It's a real problem, and it's recognized by users.

"In the case of electronic mail," Bengston said, "most of the stuff is prototype or experimental anyway."

Sophisticated Storage, Retrieval

Electronic mail includes sophisticated storage and retrieval methods along with multiple routing, according to Bengston.

"It's a series of things required to differentiate it from facsimile or the telephone. It's like data communications

and motivating people."

The controversy over analog vs. digital transmission will also be an issue in electronic mail.

"The way it is now, it's generally one or the other. Facsimile is generally analog and teleprinter is digital. You're probably going to want to switch from one to the other in a good electronic mail system because you may have diagrams that are better analog than digital.

"So the software and interfacing mechanisms, in addition to compatibility, are the major technical problems.

"It's not the development of new technology or new equipment — it's really putting a system together that's usable" that is the problem, according to Bengston.

Telephone service costs for data

rower bandwidths, Bengston said.

"The cost of transmission for a given time or bandwidth combination is expected to come down quite radically as more satellites are sent up. It looks like a cheaper method of transmitting over long distances.

"However, this doesn't solve the problem of local distribution. You're still going to use the telephone lines in the city. So if you have one plant here and another across town, you're probably going through the local telephone lines," according to Bengston.

Getting the technical bugs out of an electronic mailing system will be a step in the right direction, but it won't be the final step. Bengston said getting managers to use the systems properly will also be a problem.

"In order for an electronic mail sys-

without the use of a secretary.

"Easy-to-use equipment is essential. Keyboards for some people are not the answer. There's been talk about keyboards with whole phrases or words for some of the buttons.

"Or maybe you'll read handwriting. Ultimately you'll probably be able to transmit voice into visual. But that's a long way off," according to Bengston.

When an electronic mail system is fully operative, Bengston said, messages will come into the manager's office where they will be stored until he checks his terminal.

The system could also be made to signal the manager when an emergency message is being transmitted.

The problem with the system comes in getting the manager to check his mail.

Scientist to Develop EMS Language

By Jeffry Beeler
CW West Coast Bureau

PALO ALTO, Calif. — A French scientist who gained modest fame for his computer-aided studies of unidentified flying objects (UFO) recently turned his attention to a more down-to-earth DP frontier.

Dr. Jacques Vallee, a computer scientist and UFO expert who was portrayed by director-actor Francois Truffaut in the popular film "Close Encounters of the Third Kind," has won a National Science Foundation grant to develop the first formal representation encompassing all types of electronic message systems (EMS).

Vallee and his colleagues at Infomedia Corp. here hope their efforts will result in a common set of standards and definitions that for the first time will allow users and designers to describe precisely the behavior of word processing systems, electronic mail networks and teleconferencing systems.

Although Infomedia will shoulder most of the project burden itself, Vallee and his research director, Richard Miller, will rely heavily on the advice of a panel of outside experts, including some of the leading figures in the EMS field.

Work on the project began April 15 and will end when the $41,700 grant expires on May 31, 1980.

Vallee, who heads Infomedia, hopes the representation developed by his data communications and teleconferencing firm will serve as a technical lingua franca for the disparate elements of the EMS field.

To date, the discipline has been splintered by a kind of computer industry Babel. Groups interested in EMS have found themselves unable to effectively describe their work to each other because each party in effect has been speaking in a different technical language.

Users haven't been able to precisely explain their needs to suppliers, and suppliers in turn haven't been able to accurately describe their offerings to users. As a result, confusion has reigned and the pace of EMS development has lagged badly.

Vallee proposes to end the chaos and speed the rate of technology transfer by replacing the mishmash of EMS languages with a single, common "tongue" peculiarly suited to express data communications concepts and processes.

Astronautics & Aeronautics 17(2): 42-56, Feb 1979

Big Comsats for Big Jobs at Low User Cost

By IVAN BEKEY
The Aerospace Corp.

Three design examples show that large space systems easily outstrip ground alternatives in supplying inexpensive personal wrist-radiotelephone communications to millions, electronic business and government mail delivery, and educational TV to schools throughout the country

IVAN BEKEY (AF) directed advanced space system studies at Aerospace Corp., until recently becoming head of advanced concepts at NASA Hq. For 18 years he led long-range civilian and military space program planning studies and new-technology investigations. With Harris Mayer he contributed a landmark study of space application concepts to NASA (see Jul/Aug 1976 *A/A*). Bekey is widely recognized as having originated the space concepts for a 20th Century realization of the Dick Tracy wrist radio and the use of three-dimensional holographic images for teleconferencing. In previous positions he has been responsible for technical advice and support to the U.S. Air Force on advanced space surveillance, communications, and ballistic-missile defense. Early in his career he was employed at Douglas Aircraft and RCA in design and development of missiles, radar, and electronic countermeasures. He holds an MSEE from UCLA and is a member of the AIAA Space Systems TC.

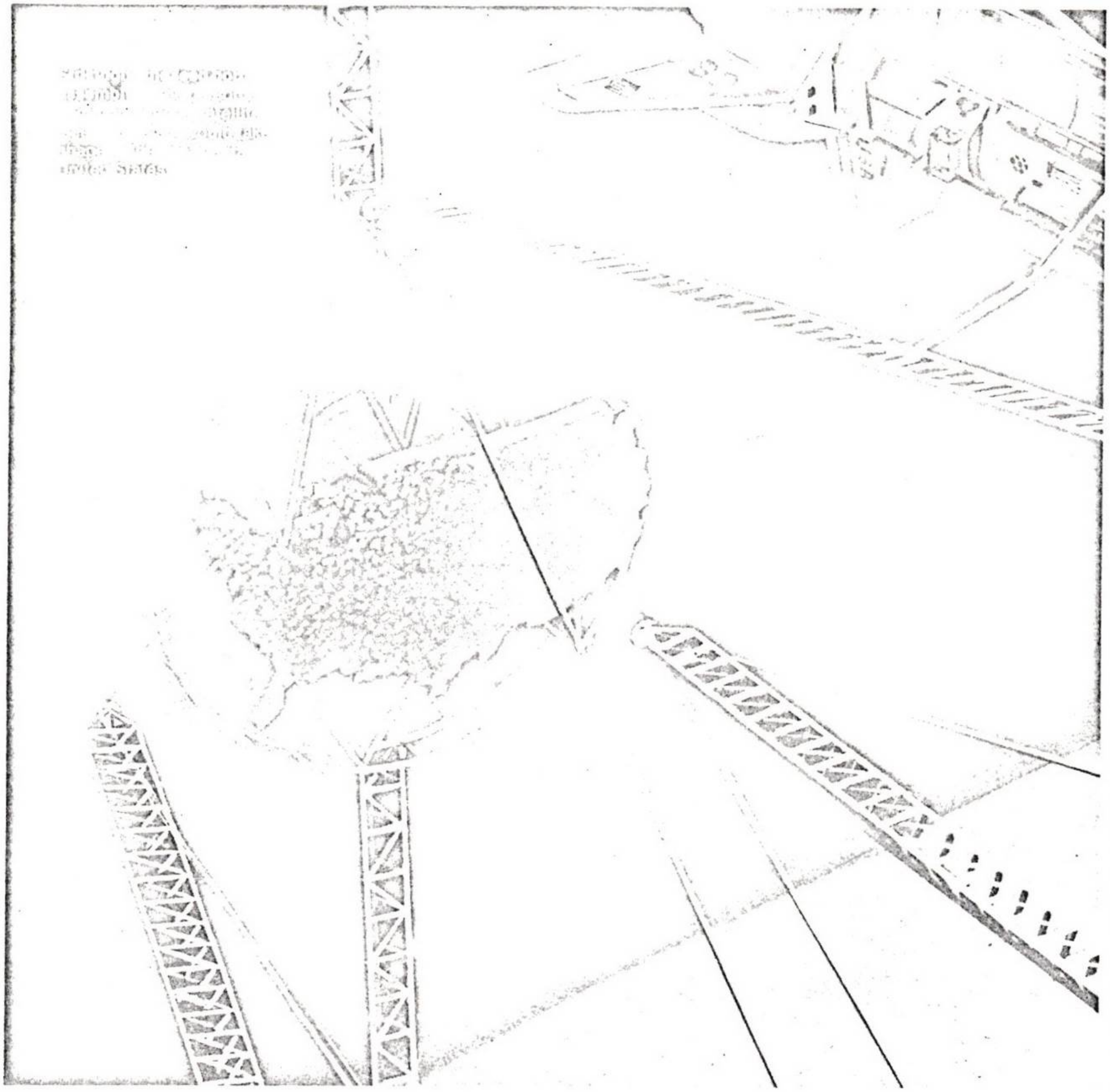
The advent of the Space Shuttle plus new technology in the 1980-90 period will allow the routine construction, assembly, checkout, and servicing of very large satellites in orbit. Such space systems will make practical a host of new services simply either not possible or economically attractive with smaller satellites.

Over the next decade we should, indeed, see a "complexity inversion" compared to past practice, particularly in communications satellites. That is, whereas it has been past practice to make the satellites as small and inexpensive as possible, paying the price in very large and expensive ground stations to communicate through the satellite, the current trends are increasing satellite antenna size and power. This in turn is reducing the power and antenna size required in the ground terminals, thus reducing their cost, and allowing more to be used. Already experiments have run color TV through ground antennas as small as 0.6 m.

In the next decade, this trend could produce satellites so large and powerful that the two-way user terminals shrink to desk-top or even wrist-watch size. This shrinking of terminals on the ground will make them very inexpensive, widely available, and extensively used—thus reducing communications via satellite to a routine matter for millions of people. Space operations will then acquire a highly visible importance in the lives of Americans, in contrast to the little-recognized use of a few large ground terminals by a few large companies now.

Very attractive system concepts have been identified by the author and other workers under contract to NASA in the past several years by applying this complexity-inversion technique (e.g., see the Jul/Aug 1976 *A/A*, p. 34). The concepts run the gamut from personal radiotelephone terminals the size of a wristwatch, and costing no more, to holographic (3D) image transmission for teleconferencing through rooftop small antennas. Rather than leading to "wired cities," these developments could create "wireless cities" and thus allow developing nations to leapfrog 100 years of technology.

To give better insight into what is possible with



large satellites, this article presents three examples drawn from work done under contract to the Director of Advanced Programs, Office of Space Transportation Systems, at NASA Headquarters. The work was done by the author while at the Aerospace Corp. and there supported by its technical specialists. The examples explore design concepts and their costs, and compare them to alternative ground-based concepts. They illustrate the extraordinary potential services space systems can give us, but should not be construed as planned NASA communications programs.

The first example concerns personal communications using wrist radiotelephones, the second, electronic transmission of mail; and the third, wide dissemination of educational TV. T-1 outlines the "requirements" imposed by the author on the space systems as an attempt to anticipate national needs for each application.

The personal communications satellite system aims at interconnecting about 10% of the 1990 U.S. population—about 25 million users anywhere in the country—direct from user to user through wrist-sized radiotelephones. The users can be mobile or stationary, and can also communicate with anyone through the fixed telephone networks via a ground-terminal entry point.

The educational television satellite system would interconnect all 65,000 U.S. schools and their 16,000 district headquarters (or all 4000 universities

T-1 INITIATIVES DEFINITION AND EVALUATION SYSTEM REQUIREMENT SPECIFICATIONS

Requirements	Personal communications (wrist radio)	Educational TV	Electronic mail
Number of users	25 million (10% of 1990 population)	65,000 schools, 16,000 districts or 4,000 univ. and colleges, 250,000 learning sites or 50 studios, 500 stations	544,000 terminals at 174,000 business offices 370,000 government offices
Location of users	2/3 urban, 1/3 rural		Corporation headquarters and major field offices, all gov't office buildings
Mobility of users	Mobile	Fixed/transportable	Fixed
Information	Voice	Color TV/2-way voice	Letters, data, and graphics
User's antenna size	Wrist set	3 ft for schools 10 ft for districts	3 ft
Access time	<1 min in non-peak hours <5 min in peak hours	1 hr/day/district (avg. shared channel)	1 hr typical 10 hr maximum
Use time	>5 uses per day, 1-min minimum per use	24 hr/day	10 hr/day 260 days/year
Terminal life	16 hr before recharging	N/a	N/a
Terminal-cost goals	$<\$30$	$<\$300$ plus TV	$<\$1000$—small users $<\$10{,}000$—large users
Safety considerations	Power density: $<10^{-4}$ w/cm^2 at 10 cm	N/a	N/a

and colleges with 250,000 remote learning sites) with color TV and interactive audio. It would use small antennas on each school and district building to disseminate program material and share programs between poor and rich districts across the country.

The electronic mail system would exchange first-class "mail" between business and government buildings (not homes) directly via satellites—in effect "delivering" about 15 billion pieces of mail per year (about 15% of the estimated 1990 total mail flow) among 544,000 small terminals, each on a separate office building.

Introducing such large-scale satellite services might have some transient adverse social and institutional impacts. The author did not attempt to assess potential impacts but believes they should be considered now. This article is limited to providing factual cost and performance data to such arguments.

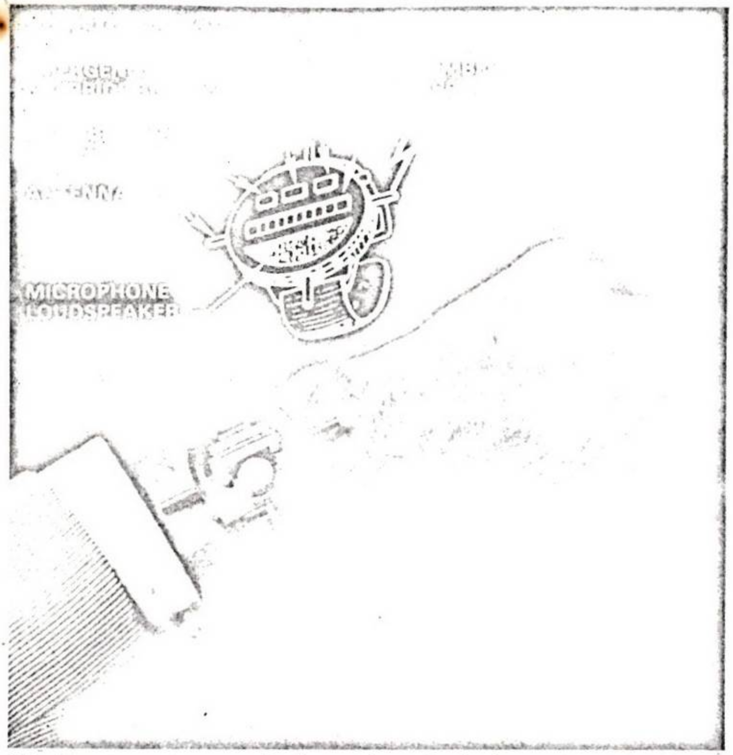
Personal Communications: The following page depicts the personal-communications system concept (F-1 to F-7). It uses a single large communications satellite to link the 25 million users outfitted with wrist-mounted radiotelephones. Due to the very small power and antenna size possible in such a radio-telephone, the satellite antenna must be large, and all channels demodulated to the audio baseband in the satellite to yield an adequate signal-to-noise ratio. A radiated power of 0.25 w through omnidirectional stub antennas was considered reasonable, with enough battery energy to support at least five 1-min transmissions per day without recharging the wrist-radio batteries.

This defined a satellite antenna at least 220 ft in diameter. Although the reciprocity of links allows the satellite's downlink transmitter power to be as low as 0.5 w per channel, the total satellite power will be large due to the large number of simultaneous users. Analysis showed that up to 108 users could time-share a channel (given certain conditions, described in a moment). Thus the 25 million users could share 230,000 voice channels. Given the average-to-peak power factors for voice the satellite needs a radiated power of 46 kw. Furthermore, since the direct interconnection of users dispenses with ground networks (to insure lower user cost), the satellite must contain the equivalent of a telephone switching center for 230,000 trunks—a large order indeed.

This number of voice channels would occupy a minimum of 2.3 GHz of RF spectrum if the satellite antenna beam covered the whole U.S. From geostationary orbit, however, the 220-ft antenna radiating at 5 GHz has a footprint of about 30 mi. diam, forcing replication of the beams to cover the entire U.S. It takes some 6084 beams (footprints) to cover the rural areas in the contiguous 48 states and 846 to cover urban areas with 25,000 people or more each. F-2 summarizes the coverage.

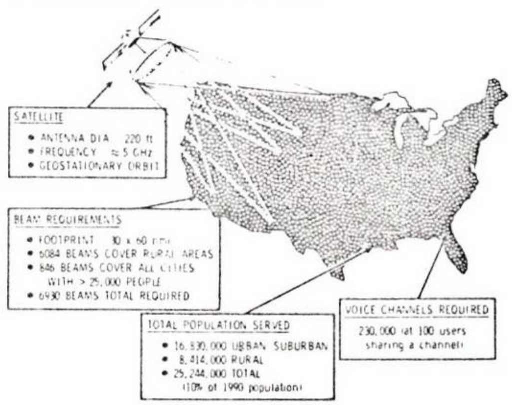
The large number of beams creates a complex satellite antenna, but it automatically solves the spectrum problem. Since non-adjacent beams can be on the same frequency, almost 7000 beams take no more spectrum than the *three* most heavily loaded ones. At 694 channels per beam maximum, and allowing for different uplink and downlink frequencies the system uses a spectrum of only 60 MHz. Lightly loaded bands at 4.7 and 4.9 Ghz could be used, or almost any bands from about 1.5 to 7 GHz.

The satellite (see F-3 diagram) has a transponder

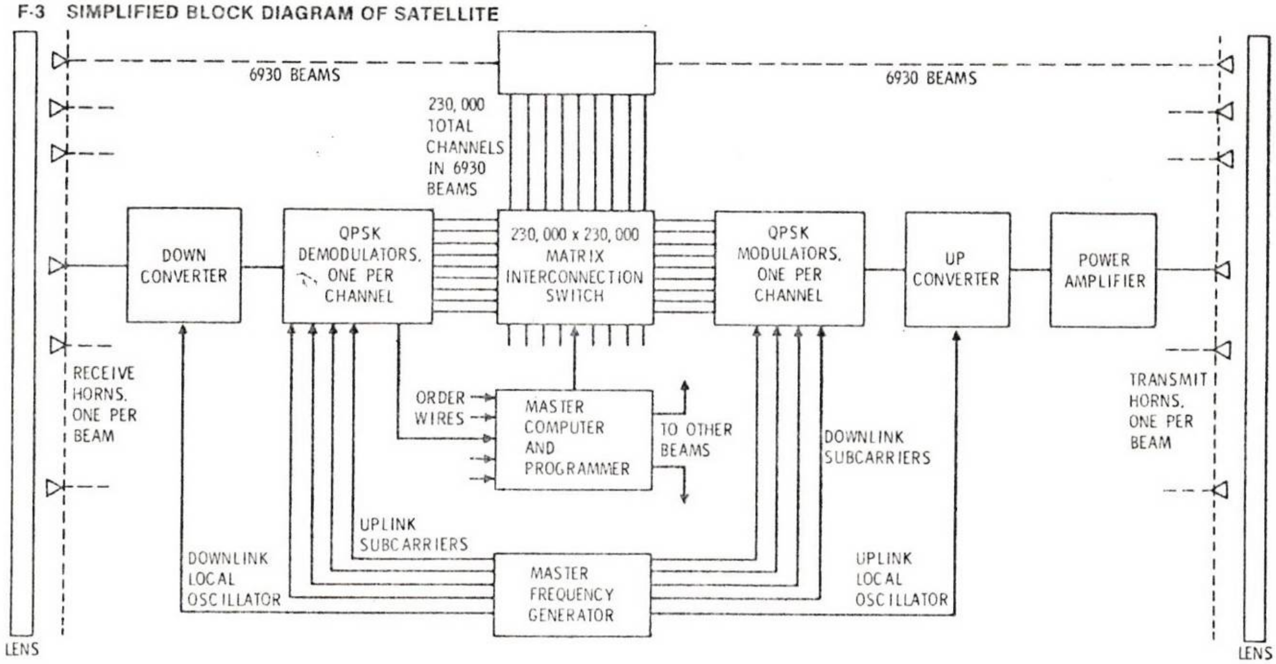


F-1 For personal communications wrist radiotelephones linked by satellite would serve many personal needs. Radiotelephone, shown here as mockup, should cost less than $10.00 in mass production.

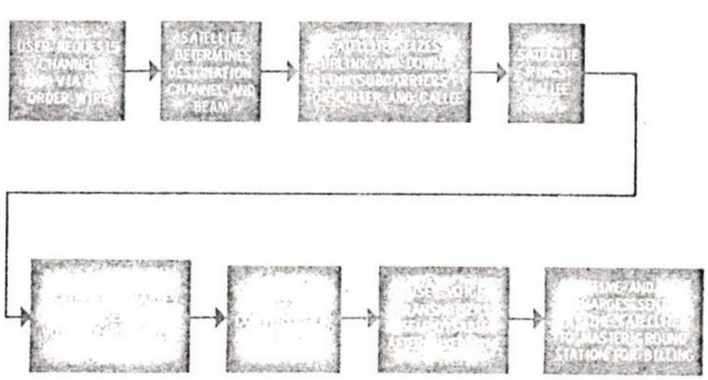
F-2 COVERAGE AND BEAM REQUIREMENTS FOR PERSONAL COMMUNICATIONS

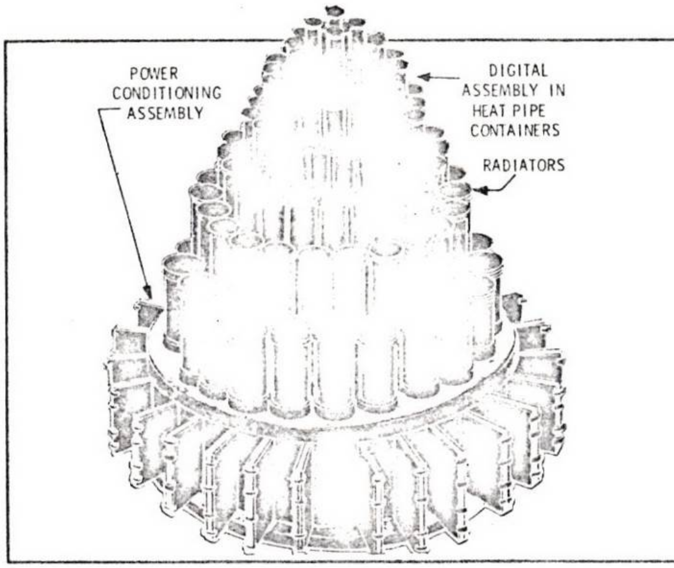


F-3 SIMPLIFIED BLOCK DIAGRAM OF SATELLITE



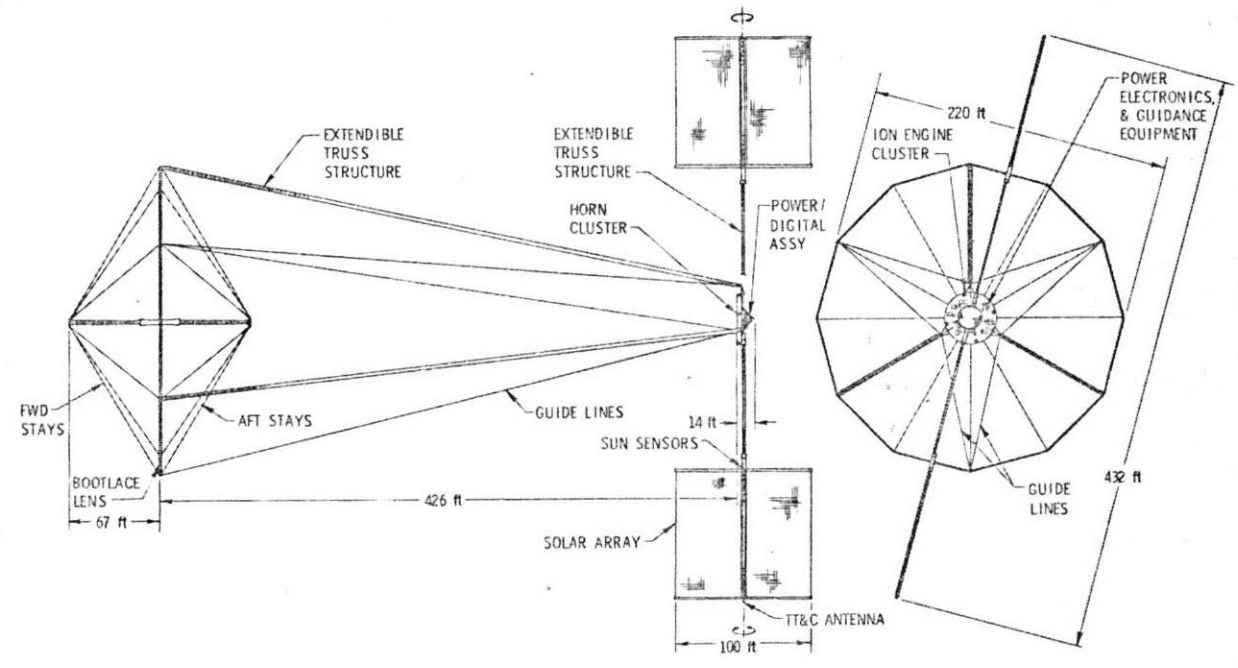
F-4 SIMPLIFIED OPERATING LOGIC AND DISCIPLINE



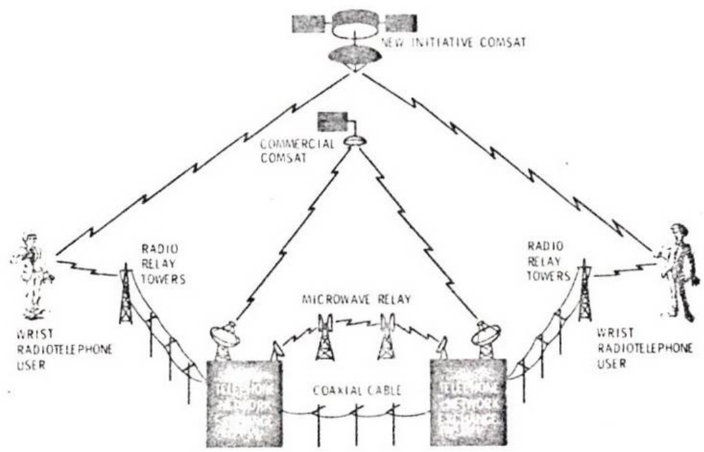


F-5 COMPLETE PROCESSOR AND SWITCH
Design: scaled No. 4 ESS; 1987 projected technology; uses wafer-scale integration; integral thermal-control system; all modular space plug-in assemblies. Functions performed: 230,000 by 230,000 voice-line exchange switching; master processor for order wires and system discipline; controller for switch; system "health" monitor; and user-charge processor. Overall characteristics: Weight, 8000 lb; size, 15-ft diam by 8-ft height; power, 50 kw.

F-6 SATELLITE PLAN



F-7 MAJOR SYSTEM OPTIONS FOR PERSONAL AND EMERGENCY COMMUNICATIONS



for each beam, each transponder demodulating the signals to baseband and providing separate voice-channel outputs to the switch. An on-board processor recognizes the addressee code of each message, and switches the signal to the proper channel to be remodulated and downlinked through the proper transmitter and beam to reach the intended user. Although the system has 6930 transponders, none need have more than 380 w of output power, and most only a few watts. All-solid-state transponders are envisioned in 1990.

The operation of the system requires a "central discipline" to avoid the chaotic interference so typical on Citizens Band radio and to allow use of such very low-power user sets. A satellite-based discipline has been designed that maintains positive control of who has access to which channel and for how long by enabling (and disabling) and tuning the user-set voice transmitters and receiver by command (F-4). The user can activate a very-narrow-band request link at any time. The satellite places his call request in a queue, "connects" the user's voice transmitter and tunes it to the proper channel when his turn comes up, and disconnects it after a fixed time. The "discipline" imposes a call duration of 1 min. Requests for longer calls receive a lower priority (during peak periods) in the queue and are charged at a progressively higher rate. Emergency calls receive immediate assignment.

The onboard switch would need twice the capacity of the Bell System ESS-4, the largest electronic telephone switchboard today. Scaling of the ESS-4 using 1987 technology wafer-scale integrated circuits would produce the relatively compact device illustrated in F-5, requiring only *one-tenth* the weight and power of the ground-based ESS-4.

The antenna design resulted in a bootlace lens in the form of a set of stretched thin films with three planes—two of dipoles and one of phase delays. The lens would be illuminated by the array of 6084 feed horns illustrated: The country-wide coverage gives it the mirror-image shape of the mainland United States.

The antenna could be deployed in space using a clever scheme suggested by Grumman Aerospace. The lens deploys from a stowed cylinder, the supporting struts elongate much as an Astromast, and guy wires support the whole against a compression-rim structure. Although the antenna could be space-assembled, it need not be.

The satellite and system characteristics may be outlined as follows:

Overall:

Number of voice channels 230,000
Number of users 25,000,000
Number of beams 6930
Coverage. . . . All U.S. except Hawaii and Alaska
Access time . . . 5 min average during peak hours, but immediate for emergencies

Satellite:

Antenna diameter . . . . 220 ft (67 m)
Number of dipoles/phase shifters in lens 46,347
Number of RF channels. 6930
Number of baseband channels 230,000
Baseband switching . . . . (230,000 × 230,000 trunks, non-blocking)
RF power per channel, kw 0.5
Largest RF power/beam 380
Total RF power, w 46
Total prime power, kw 280
Satellite weight, lb 54,000

Users:

Antenna . . . hemispheric/omni pattern with slot, stubs, or lens
RF power, w 0.25
D.c. power, w 0.6
Modulation . . . . VSDM at 14.1 kbps
Weight, oz 3
Use . . . . five 1-min transmissions per day
Battery recharge. . . . . overnight

The satellite (see this month's *A/A* cover and F-6) would weigh 54,000 lb, have a 280-kw solar-cell power system, and transfer itself to geostationary orbit (GEO) following assembly and checkout employing three Shuttle flights to low Earth orbit. The satellite electronics would be all modular to allow unmanned repair in the operating orbit, as well as manned checkout in the Shuttle orbit before transfer. The large electrical power source on board needed for communications would power ion engines to make the transfer. In final orbit, these ion engines would be rotated to serve for on-orbit attitude and stationkeeping translational control. The solar cells are oversized by some 35% to allow for degradation during transfer.

The wrist radiotelephone (mockup, see F-1) would contain a micro-electronic chip, a battery measuring about 0.25 in. on a side, a stub-crossed dipole or self-contained cavity-backed slot antenna, LED displays, and a number of controls. It should weigh no more than a large wristwatch, and be able to communicate at least five 1-min messages during any 16-hr day before recharging overnight. Such radios, mass-produced, should sell for about $10 (using current prices for electronic digital wristwatches as a model). Each would have a unique digital owner's code (phone number) built in.

When could this personal communication system begin serving the American public? Started in 1980, a program of two technology-demonstration steps could readily permit commercial operations in 1990—earlier if the program ran concurrent demonstrations and associated engineering.

To make system costs meaningful, they were compared against viable ground-based alternatives designed to the same requirements. F-7 illustrates these.

The preferred one chosen for comparative analysis comprises a large number of radio relay towers connecting the wristradio users to the fixed telephone networks, which in turn provide the interconnection and switching for system operations. The towers would cover most of the contiguous U.S., even though telephone-line connections to the towers would not normally be available in wilderness and unpopulated areas.

Thus, the coverage expected would be less than that of the space system, although the population served would be essentially the same. The switched telephone networks expected to exist in 1990 will probably be able to support the required volume of messages with but minor augumentation. The maximum range expected between wristradios and towers would be about 5 mi, thus requiring about 60,000 relay towers 150 ft high, or on high buildings, with an average of 15 w in each tower transponder.

For pricing purposes, 90% of calls made by a user were assumed to stay within his local exchange zone, with service charged at a flat rate per month. Ten percent of the calls were assumed to go outside the local zone, with service charged at current long-distance toll rates.

The same ratio of calls was used for the space system; but, it must be remembered that it is insensitive to local/distant call ratios. The system costs were determined as well as the per-call user costs.

T-2 outlines the total costs of the space and ground options over a 20-year period. For the space system, they include research, development, test and evaluation (RDT&E) of two demonstration and one operational systems, all investments in the operational satellites, all user and ground equipment, and operations and maintenance of the final satellite system, the ground system, and terminals for a 10-year period, including one Space Shuttle and IUS flight per year to service the satellite in GEO. These costs also cover operating and revising the demonstration and test satellites through 1991, the RDT&E for the wrist-radio sets, and the wholesale price of 25 million wrist radiotelephones.

It was assumed that a single agency or entity would undertake all phases and cover all of the costs, including ground and space segments. This was done so that the total cost of operating the system could be weighed against expected income. It should not be construed as a recommendation for any financial, institutional or government/industry arrangement.

The whole space system would cost $2.3 billion, while the ground alternative (computed in the same way) would cost $20.7 billion. By way of comparison, for the same number of channels, mix of local and long-distance calls, and 10 years of operation, the telephone system today charges about $13.8 billion. That is, more than half of the cost of the ground alternative arises from telephone-network service charges. Note, too, that the satellite system would have somewhat wider coverage than the ground alternative.

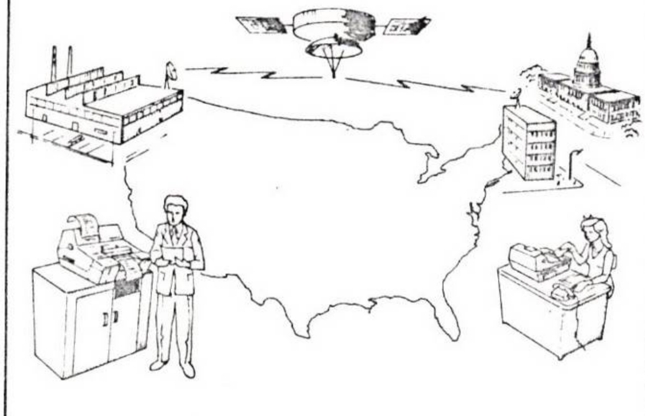
The fourth column in T-2 shows the call costs to the user (computed by dividing the total 20-yr cost by the number of minutes during the 10-yr period that the system is expected to be operating, reduced by a factor of 37% to account for the average loading expected in the system channels). Using these assumptions, *the total R&D, investment, and operation costs in the satellite approach could be recouped in a 10-yr period by charging the user only half a cent per minute of conversation.* The alternative ground system would have to charge 5.4

T-2 COST COMPARISON FOR PERSONAL/EMERGENCY COMMUNICATIONS

System alternative	Coverage	Total 20-yr costs, $ billions [b]	Call costs, [c] cents/minute	Net annual profit or loss from operations charged at 13.3 cents/minute
Space	Complete all USA [a]	2.29	0.5	Profit of $1.6 billion
Ground	Incomplete in rural/wild areas, underpriced, or both	20.76	5.4	Loss of $0.4 billion
Current telephones (not mobile or portable)	Fixed point coverage only	13.8 (service only)	3.6	0

[a] Not including Hawaii and Alaska. [b] 10-yr RDT&E; 10-yr operations and maintenance; all user equipment. [c] Assumes 37% average channel loading.

F-8 ELECTRONIC MAIL TRANSMISSION



cents per minute. (Today's telephone networks would charge 3.6 cents per minute. This number goes up with an increasing ratio of long distance to local calls. In contrast, the space system charge of 0.5 cents per minute would be independent of that ratio.)

These costs to the user represent the breakeven charges (entire RDT&E, investment, and 10 years of operation and maintenance paid off at the end of the 10-yr period). They assume no profit and no cost of capital.

What might reasonably be the profit or loss in these systems to private industry if it charged users at a rate competitive with today's regulated telephone system? The last column in T-2 gives the answer. Under the assumption that the users are charged an average of 3.6 cents per minute of conversation, the ground system shows a loss of $0.4 billion a year, while the space system would generate a net profit of $1.6 billion a year. The space system would have an income of $16 billion over the 10-yr operations period—a net return of 70% on the investment in the first year of operation and for every year thereafter. These numbers become even more impressive with more long-distance calls, the likely trend once the users discover that long-distance calls do not cost extra. Will industry pass up such an opportunity?

Electronic Mail: F-8 depicts the system for direct exchange of electronic mail among business and government users. The T-1 goals result in 540,000 user terminals—one each in every building complex of corporations with assets greater than a million dollars (the study cutoff) and government office buildings with more than five permanent people. Most of these terminals would be situated within a 50-mi. radius of the 846 cities having populations greater than 25,000 (F-9). By 1990 the business-government sector of the mail stream should total 15 billion mail pieces per year.

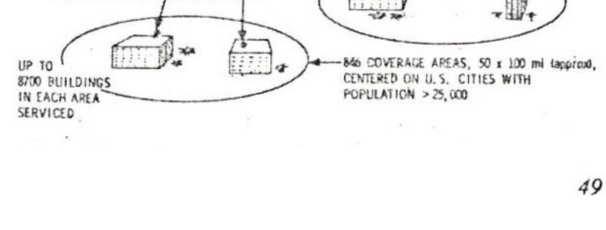
Reading and reconstructing a piece of mail takes about 405,000 bits on the average. The data rate required to make a total mail exchange once per day would be suprisingly low: 452,000 terminals would require only a 0.5-kbps terminal and facsimile machine. In addition to facsimile, many larger offices might install optical character readers and direct typewriter/magnetic-tape inputs, and operate through their computer center. This study does not assume so advanced a practice.

This system also employs a multibeam antenna, but with "only" 846 beams. All users within the footprint of a 50 mi.-diam beam would contact each other directly through a transponder for that beam in the satellite, using a time-slot TDMA scheme, while users in different beams would make contact during appropriate time slots through a buffering processor in the satellite. F-10 describes the typical time sequence. All messages would be digital and all would include an addressor-addressee code. The basic frame time during which all users would be sampled was chosen as 10 minutes. The largest users would send or receive at least one letter. Each terminal would have 10 min of buffer storage.

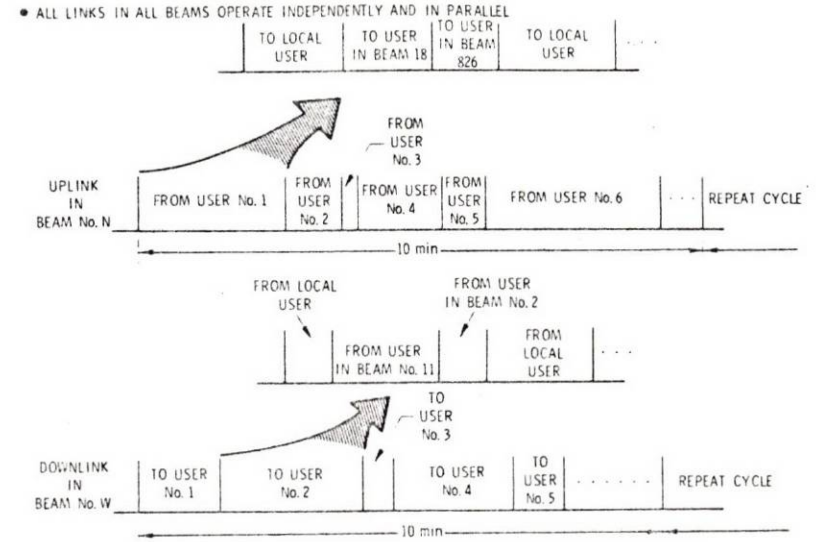
F-11 shows the satellite block diagram. Each of the 846 transponders would have an output power of 3 w and a modulator and demodulator to baseband. The satellite would have an antenna similar to the one for the wrist-radiotelephone system, except for the number of feeds; it would operate at 20 and 30 GHz. This frequency selection was based on ready allocations and mail delivery not having to go in real time in the face of outages due to rainfall attenuation. The satellite and terminal buffers would be sized to accommodate the expected outages.

The antenna would measure 9.5 m in diam at 20 GHz and have a 30-GHz inner section of 6.4-m diam with dual-sized elements. Each of the 846

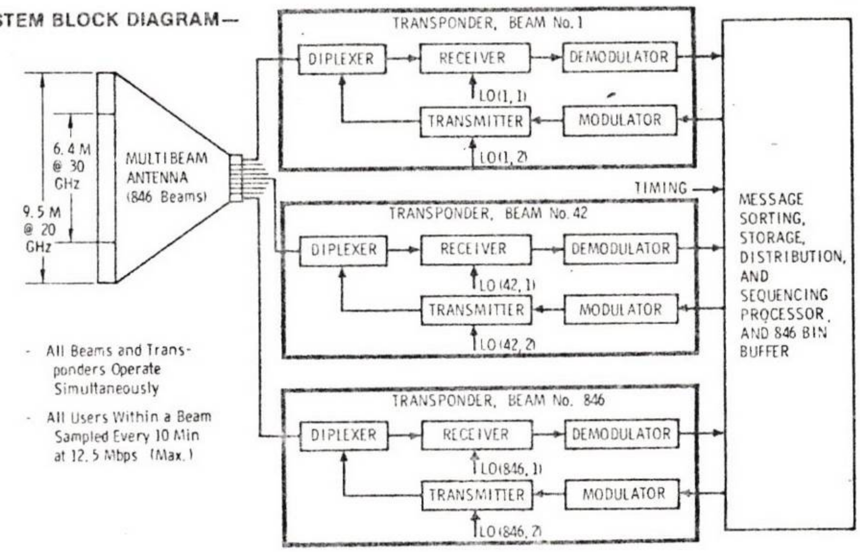
F-9 SATELLITE CONCEPT FOR ELECTRONIC MAIL



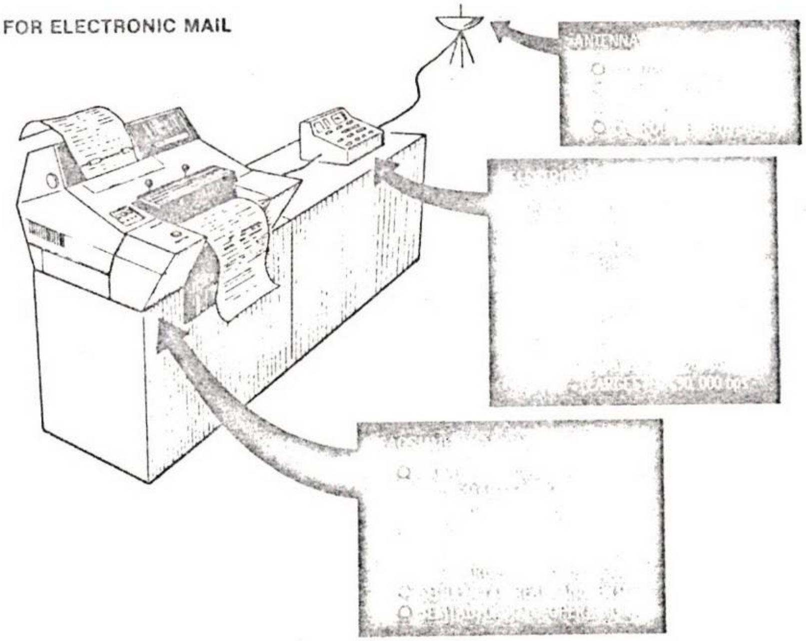
F-10 ELECTRONIC MAIL COLLECTION AND DISTRIBUTION



F-11 SATELLITE SYSTEM BLOCK DIAGRAM—ELECTRONIC MAIL



F-12 USER TERMINAL FOR ELECTRONIC MAIL



beams would cover an area about 50 by 150 mi. The most heavily loaded beam would occupy 12 MHz of bandwidth. Again, due to extensive frequency reuse, the total spectrum requirements are less than 75 MHz.

The satellite processor, essentially an 846-bin buffer, would receive "letters" into each bin from users in other beams, to be downlinked through each respective transmitter. No permanent storage would be needed, since the average input and output rates are equal.

F-12 illustrates possible ground terminals. They would come in four sizes, all operating through a 3-ft fixed antenna on the roof. Facsimile machines would be used for the input and output devices at every terminal—very similar to today's commercially available units which operate over a phone line and transfer one page every 2-6 min. One facsimile unit capable of sequential reading and printing operations would be required in each terminal. Units would send and receive "mail" once every 10 min, reading and printing all daily letters at least every 10 hr. Both graphics and type would be handled. The capacity of the terminals would vary from 31 to 2620 letters per day.

The system major characteristics may be outlined as follows:

Satellite segment

—Single: 5300-lb satellite in GEO.

—Antenna: 9.6-m; 846 beams; footprint, 50-by-100-mi. each.

—Solid-state transponders: 846 independent, 3 w each.

—Frequencies: downlinks, 20.0-20.1 Ghz; uplinks, 29.5-29.6 GHz.

—Bandwidth: 12-MHz maximum per beam; 75-MHz total up and down.

—Solar-cell power: 13 kw.

—Switch: all-digital 846-channel store and forward; 400 lb.

—Propulsion: self-powered ascent to GEO using ion engines.

User segment:

—Fixed antenna: 3-ft diam.

—Transceiver: 6.3-w transmitter; uncooled receiver.

—Bandwidth: 12 MHz maximum.

—All users sampled every 10 min.

—Buffer store for 10-min mail volume.

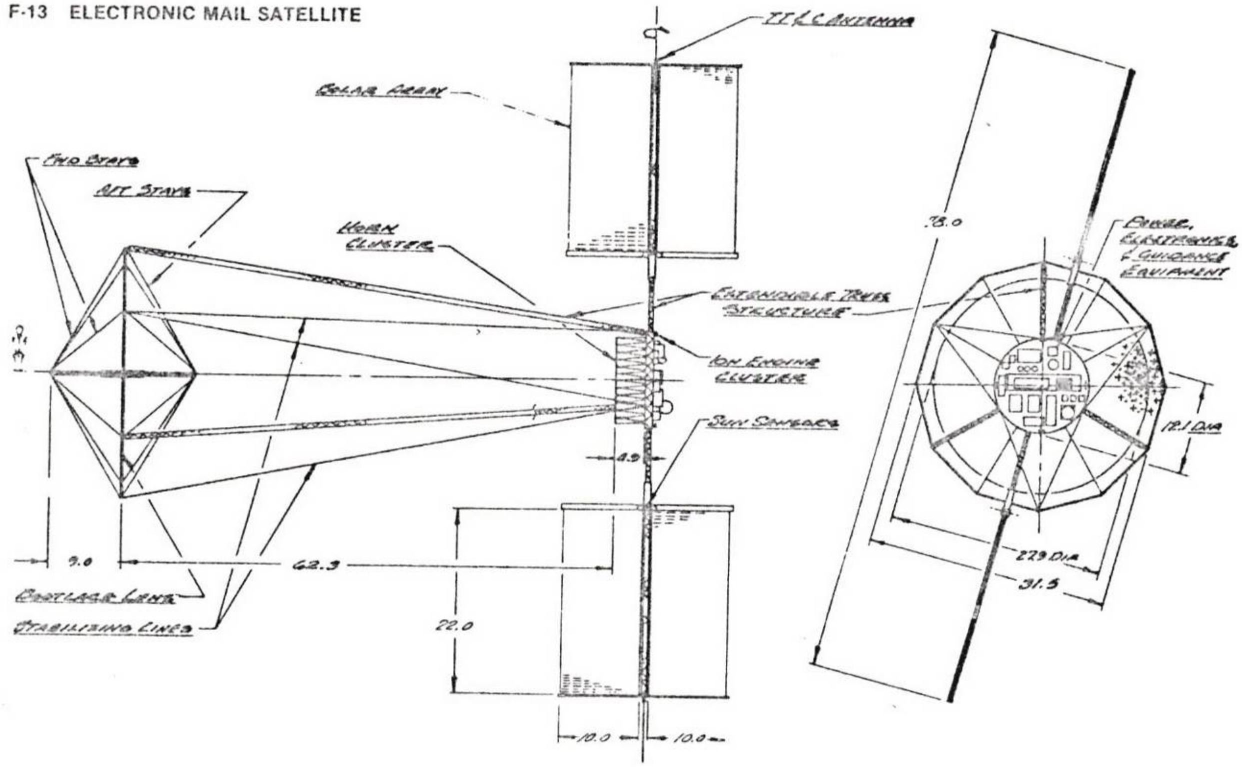
—Smallest terminals: 452,000 4-min/page fax machines delivering 31 letters per day each.

—Largest terminals: 5360 4-page/min fax machines delivering 2620 letters per day each.

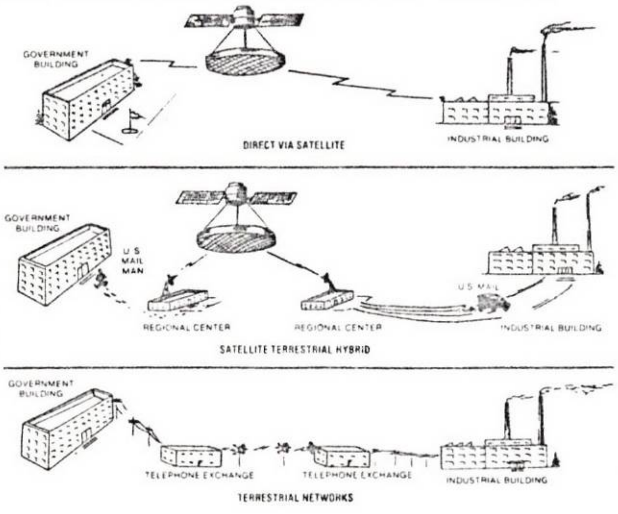
—Total capacity: 15 billion letters per year.

The satellite (F-13) would weigh 5300 lb with its 13-kw solar cell power supply. Launched into LEO in a single Shuttle, it would transfer itself to GEO with ion engines. (Solar cells would be oversized by

F-13 ELECTRONIC MAIL SATELLITE



F-14 MAJOR SYSTEM OPTIONS FOR ELECTRONIC MAIL



35% to compensate for degradation during slow ascent to GEO.

This satellite could be readily launched in 1988 from a 1980 start, with only one technology-demonstrator precursor. It would be checked out in orbit before transfer, then serviced in GEO by teleoperator.

Alternatives to this space system (see F-14) include conventional physical pick-up and delivery of letters by mail carrier, with physical transport between post offices; a hybrid system in which a number of collection and distribution centers process the mail for electronic exchange, but still depend on physical pickup and delivery, and an all-electronic system operating through wire building connections and the normal telephone switching and transmission system. The last makes the most revealing comparison. But the USPS recently analyzed hybrids, and its results will be also used for comparison.

To some extent the current postal service and the hybrid option are "apples," whereas the space and ground alternatives defined in this study are "oranges." That is, this study treats only the government and business first-class mail and did not treat the home segment. On the other hand, all the options and the current USPS exchange 15-20 billion pieces of mail per year, and this fact makes the comparison reasonably fair. (The mail costs to the government are assumed chargeable on a per-letter basis rather than on an average of each agency or department use. Since patterns of mail usage develop rapidly and stay very stable, there is no significant error from such assumption.)

Thus we have the figure in T-3. The second column gives the total, 20-yr cost of each option. As before, this cost includes all user equipment, the 10 years of RDT&E and investments plus 10 years of operations and maintenance for space and ground segments. It covers one maintenance flight every three years for the space alternative plus purchasing, operating, and maintaining the 540,000 terminals The whole space system thus costs just under $4 billion.

For the ground alternative the same assumptions have been made, except that costs charged by the telephone networks are included as part of the operating costs, rather than satellites. These costs are charged to the system at *half* the amount required by the regulatory agencies to remove the current regulatory overburden and make the most favorable comparison for ground concepts. These costs were taken to be $12 per month flat rate for local mail and 17 cents per min for long-distance "mail" transmission per terminal. Even then, the total costs are a whopping $38.5 billion.

The Postal Service's hybrid was chosen from among several studied. It includes 360 regional collection and distribution centers interconnected via commercial communications satellites, an annual volume of 22 billion mail pieces, 6220 public terminals, and mail-carrier delivery. It would cost $5.8 billion for 10 years of operation, including all development, test, and evaluation of the equipment to transfer mail from center to center. In addition, the cost of the mail carriers for 22-billion mail pieces per year, at an average of 7 cents a letter, would total about $15.4 billion over 10 years. The grand total: $21.2 billion.

The third column of T-3 shows the equivalent costs per letter (total costs in the previous column divided by the number of mail pieces carried in 10 years). The space system amortizes all of the costs over this period and breaks even at only 2.6 cents per letter. The all-electronic ground alternative would cost almost an order of magnitude higher at

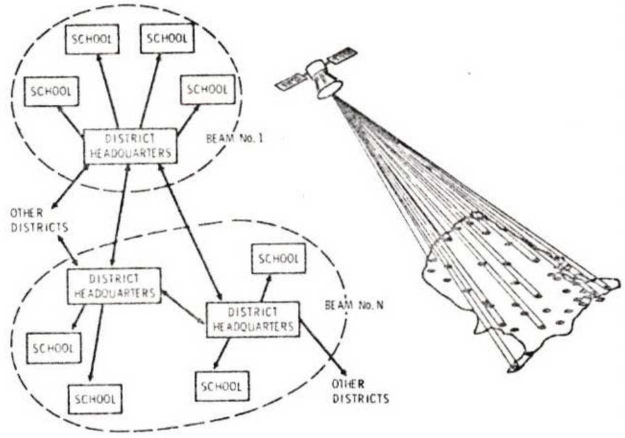
T-3 USER COST COMPARISON FOR ELECTRONIC MAIL
Expressed in 1977 dollars, and including all R&D, investment, ten years of operations, and user equipment. Equipment costs prorated over ten years. Assumes 15 billion mail pieces per year for the space and ground alternatives, 22 billion for the USPS hybrid, and about 70 billion for current USPS.

System	Total 20-year costs, $ billions	Costs per letter, cents
Space initiative 540,000 terminals	3.96	2.6
Ground alternative 540,000 terminals	38.5	25.3
USPS space/ground hybrid 360 distribution centers with mail carrier, P/U-deliv. plus 6220 public terminals	5.8 center-center 15.4 mail carriers 21.2 *Total*	2.9 center-center 7.0 mail carriers 9.9 *Total*
Current USPS	—	15.0

25.3 cents per letter. The Postal Service hybrid shows almost equal costs to the space system for the center-to-center (electronic/satellite) portion at 2.9 cents per letter, but the cost of the mail carriers for the bulk of the mail must be added. Since this averages about 7 cents per letter, the hybrid totals 9.9 cents per letter. Today the Postal Service charges, of course, 15 cents per letter. It is clear that electronic mail using either pure space concepts or space/ground hybrids, offers less expensive service than physical handling. The pure ground alternative drops from the competition, however, primarily due to the charges of the telephone network (which would be approximately double if the currently regulated charges had been used).

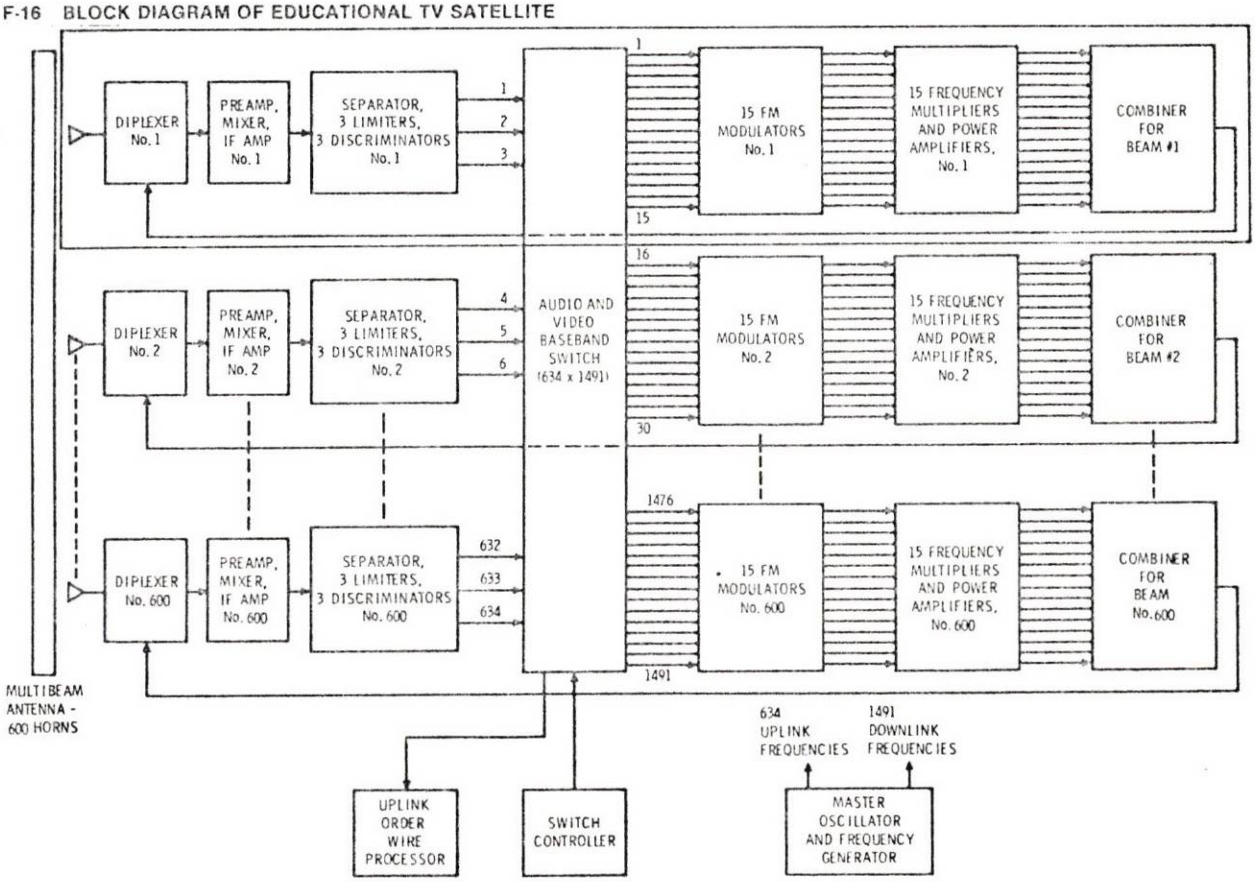
The charge of 9.9 vs. 2.6 cents per letter for the hybrid and pure-space options respectively, needs further explanation, as the two do not represent quite the same class of service. The hybrid would handle all classes of mail including home delivery, but would not collect or deliver mail with the timeliness of the space approach, which would collect and deliver mail every 10 min. The space system also would offer the convenience of being able to send mail directly to and from the user's office building rather than requiring delivery to/from post offices as in the hybrid approach.

F-15 SATELLITE CONCEPT FOR EDUCATIONAL TV
Covers U.S. by 600 areas, 50 mi. diam each, which include all cities/towns > 25,000 population. One-way TV from districts to schools. Two way audio in schools. Two way TV among districts. Reaches 16,000 district headquarters and 65,000 schools. All connections direct through satellite.

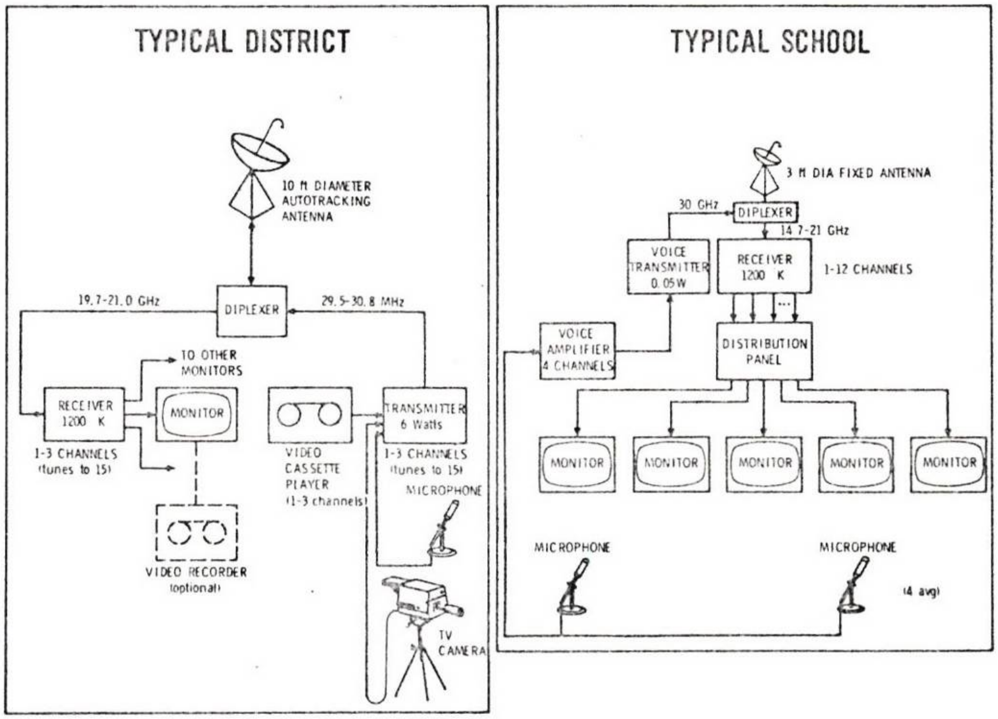


On a strict dollars-and-cents basis, then, the pure space system appears to be by far the most attractive choice. Institutional and social questions regarding job displacements and the likely attitudes of the labor unions to the subsequent retraining of the work force, initial phasing for gradual service

F-16 BLOCK DIAGRAM OF EDUCATIONAL TV SATELLITE



F-17 EDUCATIONAL TV TERMINAL



introduction, all-inclusive vs. limited business/government service; and how to handle packages and other classes of mail, go beyond the scope of this study. It seems evident, however, that this type of service will come eventually, and that space offers the key to the lowest user costs coupled with best service.

Educational TV: An advanced educational TV concept with large-scale distribution of program material to all U.S. schools, and with the aims given in T-1, would require the satellite to transmit almost 1500 TV channels simultaneously through 600 beams, each covering an area of 50 by 100 mi. on the ground, centered on populated areas within the contiguous U.S. In addition, it would require 634 simultaneous uplink TV channels from the districts for program origination. Complete freedom of connection must be established by onboard switching so any district could broadcast programs to its schools, to other district schools, to all schools nationwide or other, in-between options. In addition, any 10% of the 16,000 districts must be able to interchange or share 10% of their program libraries in real time through the satellite to make the best material available to all districts regardless of their financial resources.

The system concept evolved (see F-15) in many ways resembles the other two systems described in this article. Schools would be outfitted with receive-only TV and two-way voice for interactive instruction. Districts would have two-way TV and voice.

F-16 shows a block diagram of the satellite. It consists of a 600-horn multibeam antenna and 600 transponders demodulating to baseband. All TV channels would be switched by a crosspoint diode matrix on the satellite. Its connection flexibility would vary all the way from a one-channel nationwide broadcast from one district source, through three different channels sent over each beam by 160 districts with recipient schools located anywhere, to 634 districts originating different programs simultaneously, which are shared by no more than three down channels each. These and other, in-between combinations allow highly flexible program sharing and distribution in response to commands for on-board switching from a ground-control center.

The system was designed to operate at 20 and 30 GHz, in this case to accommodate the extreme bandwidth required: 1620 MHz, even with the large frequency-reuse factors attained with the multibeam antenna. (The bandwidth required would be 80 GHz without frequency reuse and could not be available at frequencies much below 800 GHz.) The necessary satellite transmitter power was found to be 10 w per channel, or a total of 15-kw RF. The antenna diameter measures 9.6 m at 20 GHz, as in the electronic-mail satellite.

This satellite could transmit programs to 3-ft fixed antennas at the schools, and receive them from 10-ft tracking antennas on district buildings with 6-w transmitters (see F-17). The terminals would distribute TV to half of the classrooms in the U.S.—about ten classrooms per school. The schools would send the two-way audio through a 50-mw transmitter.

The major system characteristics may be summarized as follows:

Satellite segment

—Single 9800-lb satellite in GEO.
—Antenna: 9.6-m diam; 600 beams; footprint of 50 by 100 mi. for each beam.
—Solid-state transponders: 600 independent; 15 w each.
—Frequencies: downlink, 19.7-21.0 GHz, uplinks, 29.5-30.8 GHz.
—Bandwidth: 540 MHz per beam; 1620 MHz total.
—Solar-cell power: 65 kw
—Switch: 634 × 1491 crosspoint PIN-diode type.
—Propulsion: self-powered ascent to GEO using ion engines.

User segment

At 16,000 district headquarters—
—Antenna: 10-ft autotrack.
—Transmitter: 6 w.
—3-12 channel TV receiver
—Video cassette recorder and player
—Microphone and loudspeaker.
—Total capacity: 1491 TV channels.

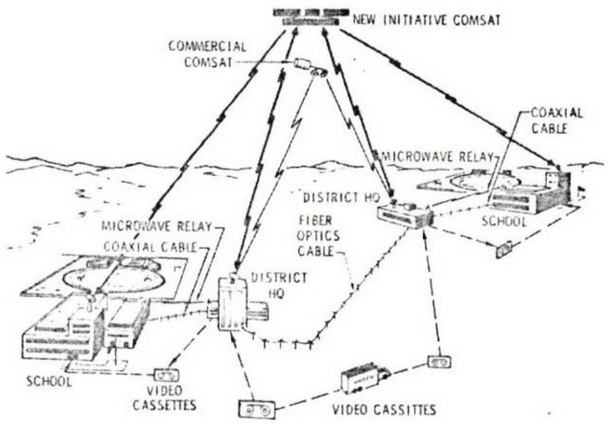
At 65,000 schools—
—Antenna: 3-ft fixed.
—Receive only TV.
—Receive and transmit audio.
—1-12 channels.
—Distribution panel.
—Average of 10 TV monitors and microphones per school.

The satellite would weigh about 10,000 lb. It would require a 65-kw solar-cell power source, also oversized to allow ion-thruster ascent to GEO. A single Shuttle flight could deliver the satellite to LEO. The satellite, as the others, would be designed for modular servicing in GEO.

F-18 shows the major system alternatives for the educational service cited in T-1. They include options for district-to-school distribution via coaxial cable or trucking of video cassettes; and district interconnect via commercial comsat, fiber optics, or trucking of video cassettes.

The distribution of TV programs from districts to schools via an average of four coaxial cables per district, and program sharing between districts using fiber-optic cables with 150-TV-channel capacity seems the most attractive of the ground options. (Commercial comsats for the 150 TV channels proved more expensive, as did trucking of video cassettes. In addition, cassettes do not allow real-time broadcasts or program sharing.) All switching would be done at the district terminals by a narrowband order wire, thus avoiding optical switching. By design, both ground and space alternatives would have equal performance.

F-18 MAJOR SYSTEM OPTIONS FOR EDUCATIONAL TV DISTRIBUTION



What would the user pay in terms of total system cost per school per year, cost per classroom hour, or cost per pupil hour? These costs were computed assuming 6 hr a day for instructional programming, five days a week, for nine months out of the year, with 30 pupils per class and half the classrooms outfitted (or about 10 rooms per school) in each of the 65,000 schools in the U.S.

T-4 gives the resulting cost comparison. The space approach proves the least expensive by any one of the three measures. To provide a frame of reference for the order of magnitude of the costs, both are compared against the average cost for an

T-4 USER COST COMPARISON FOR EDUCATIONAL TV

System	Cost per[a] school per year, dollars	Cost per classroom hour, dollars	Cost per pupil-hour, cents	Total[b] 20-yr costs, $ billions
Space initiative	4,200	0.36	1.2	2.73
Ground alternative	20,800	1.78	5.9	13.53
Teachers	≈ 116,000	≈ 10.00	≈ 33.3	≈ 75.4

[a] Average of: 10 classrooms per school equipped for ETV; 6 hr/day, 5 days/week, 9 mo/yr; 30 pupils/classroom.
[b] 10 yr RDT&E; 10 yr operations and maintenance; all user equipment.

T-5 OVERALL COST COMPARISON

Alternative	Electronic mail	Educational TV	Personal and emergency communications
Space	2.6¢/letter	$.36/class hr	0.5¢/minute
Ground	9.9-25.3 ¢/letter	$1.78/class hr	5.4¢/minute
Current services	15¢/letter (stamps)	≈ $10/class hr (teachers)	3.6¢/minute (telephones)

average teacher, which is probably in the order of $10 per hour without administrative burden. For the 10 classrooms having simultaneous instruction, the equivalent cost of teachers per school per year would be at least $116,000 as compared to $4200 for the space and $20,000 for the ground alternative.

T-4 also lists total costs of the space and ground systems over a 20-yr period. As before, these include all RDT&E and investments in satellite, user equipment, and TV monitors, and 10 years of operations and maintenance with a maintenance flight every two years. The space alternative is the least expensive, costing $2.7 billion compared to $13.5 billion for the ground alternative, and a whopping $75.4 billion for equivalent teachers.

None of the costs in T-4 covers preparing and updating a video tape library, in which most of the educational TV material would be vested. The programming figures assume that producing an average educational TV hour will cost about $10,000. It was also assumed that an initial library of tapes will be prepared that will be good for ten years and be extensive enough to allow a different program for every hour of a 6-hr day, 5 days per week, 9 months per year, for each of the 10 classrooms in a school, with 1000 school districts in the U.S. sharing each program. Furthermore, it was assumed that all schools in any given district would share programs that would permit 16 different courses to run simultaneously in the U.S. every hour of the day in a 9-month year for 10 years. Such an intital library would therefore cost $1.86 billion—or a one-time cost of only $28,600 per school.

While these costs indicate that TV education via space costs much less than by ground transmissions they do not necessarily imply a preference for TV over live teachers. In fact, since the space TV concept would cost a school less than 5% of a teacher's salary the teachers could use TV extensively as an aid at little additional expense.

Conclusions: Space systems could provide major improvements and wholly new communications services at much lower costs to the user (see the summary in T-5) than either terrestrial alternatives or current services. This is the direct result of making satellites deliberately large and complex in order to make the user terminals small and cheap. It is also the result of tackling big jobs which can take advantage of small, inexpensive terminals.

In an inflating economy it is important to compare the options using inflated dollars and present-value discounted dollars. Both comparisons show the space approaches way ahead of either terrestrial alternatives or current services.

This study could only address three out of several dozen concepts for satellite services, many of which could be as attractive as the three selected, or even more so. Further studies should examine these and other system concepts in at least equivalent depth.

A corporation or consortium should be able to turn the space systems described in this article to handsome profit. Thus, funding and development by the private sector might be expected. The roles of government and private sector in funding the demonstration of the needed technology needs to be addressed.

With properly funded programs, these systems could be operating in less than a decade. Thus, it is extremely important to examine social and institutional questions, which will undoubtedly be raised before the systems can be approved —questions concerning job dislocation and retraining of people displaced by the introduction of the new services, how to start services so that they become extensions, rather than replacements, of current work and thereby avoid great opposition; the roles of government and industry in the needed R&D; the roles of various agencies in R&D and regulation. Indeed, we need to set long-range goals for our society so that proper priority can be given such programs.

None of the designs, costs, characteristics, performance, or any other attributes of the three system concepts in this study should be considered final or optimum in any sense of the word. They represent preliminary design concepts that seem reasonable; but they have not benefitted from deep tradeoff analyses. Doubtless, many other configurations are possible, and some might be considerably better in a number of attributes than the configurations discussed here.

Furthermore, the requirements given in T-1 were self-imposed after some discussion with the using community, but without opportunity for much feedback. Quite possibly the best requirements have not been chosen for the first development item, or the likely time period in which it is to operate. Follow-on studies should examine such questions.

Above all, it is hoped that the communities and agencies that would be involved with such satellite services will benefit from this work, and cause the relevant social and institutional questions to be aired thoroughly in public forums.

Acknowledgment

My special appreciation to the technical staff of the Aerospace Corp., who helped with the analysis; to R. Ragar, who painted the cover illustration; and to R. Guest and C. Tiffany of ATT, who helped to configure the terrestrial options. ■

Telecommunications

How to write your comrade — electronically

By Mark D. Berniker
Journal of Commerce

Electronic mail is lifting superpower relations to a new level.

In the spirit of improved U.S.-Soviet relations, a California firm has established a dedicated electronic mail link that is enlarging information exchange between Americans and Soviets.

San Francisco/Moscow Teleport Inc. has created an electronic mail system with the Institute for Automated Systems of the Soviet Union.

This paves the way for electronic communication between business people, scientists, educators, journalists and others on a daily basis.

The electronic mail system will "improve the communication between representatives of different joint ventures and non-governmental organizations," Oleg Smirnov, director of the Institute of Applied Automatic Systems, said in the newspaper Sovietskaya Rossia.

A San Francisco firm has set up a dedicated electronic mail link that is enlarging the exchange of information between Americans and Soviets. Users have access to information previously unavailable in the West.

Electronic mail uses electronic signals among interconnected computers in place of paper correspondence.

The system operates off any personal computer with a modem.

That is a device that allows computers to send digital, computer-comprehensible information over conventional, voice-oriented telephone lines.

U.S. commercial users of the service are charged a flat fee of $200 a month and 25 cents for each minute they are on line with the Soviet Union.

The connect-time charge from the Soviet Union to the United States is 42 cents a minute and 74 cents for each thousand characters.

The monthly service fee for messages being sent out of the Soviet Union is 10 rubles — about $16 — to be paid in hard currency to the Institute for Automated Systems.

Joel Schatz, president of San Francisco/Moscow Teleport, said users of the electronic mail system will have access to Soviet data bases with extensive information previously unavailable in the West.

In virtually all cases, electronic mail service is far less expensive than telex transmissions. Most of the time it also is less expensive than facsimile transmission.

Any two computers equipped with a modem and appropriate software can send and receive copy in both Roman and Cyrillic alphabets.

Last year, the Soviet Union began a program targeted at implementing electronic data interchange. EDI is computerized communication using strictly standardized digital versions of common business documents.

The Soviet EDI project is called Ediflot. Reportedly, 70 computer programmers have been assigned to the effort. EDI messages can be transmitted over a network such as Teleport's.

Computers are more sensitive to bad phone lines than are people. Many users say the Soviet phone system is not conducive to computer transmission.

They say facsimile transmission, which operates at relatively high speed, is often difficult to achieve. Even Telex connections, which are relatively slow, are frequently difficult to get.

Phone service from the Soviet Union can also be expensive. A connection from Moscow to America can run $10 a minute.

To avoid difficulties that have crippled past data transmission services, Teleport is using special error-correcting modems, Schatz said.

Supposedly, these modems can recognize stray or missing characters when poor phone connections occur.

All users have to do is dial their local telecommunications company and connect directly with IDB Communications Group Inc. of Los Angeles at its Staten Island, N.Y., facility.

There the signals are picked up and sent by Globenet Inc. of Alexandria, Va., to San Francisco/Moscow Teleport's electronic mail system through fiber optic lines.

Once they are connected with the main station in San Francisco, users then type in identification codes and passwords.

Electronic mail users are given digital mail boxes when they sign up for the service. When they sign on, they can check their electronic mail just as a homeowner would pull letters out of a mailbox.

Response is simple, even when the recipient is in Siberia.

Loading messages from user computers into their electronic mailboxes to be sent takes only a matter of seconds. Messages are then dumped into the recipient's electronic mail box.

The agreement between San Francisco/Moscow Teleport and the Institute for Automated Systems is for five years.

Participants on the American side include Combustion Engineering Inc. of Stamford, Conn.; Ben & Jerry's Ice Cream Co. of Waterbury, Vt.; McGraw-Hill Inc. of New York; and Addison Wesley Inc. of Reading, Mass.

A direct link has already been established between the fusion scientists at the Stanford Linear Accelerator Center and the Institute for Nuclear Physics in Novosibirsk.

Steve Kaufman

. . . is on vacation.

Bill Zaumen

This document was produced by System Development Corporation in performance of Contract DCA100-80-C-0044

TM a working paper

System Development Corporation
2500 Colorado Avenue • Santa Monica, California 90406

series base no./vol./reissue	TM- 7038 /207/ 00
author	Mary M. Bernstein Mary M. Bernstein
technical	G. Simon Gerald A. Simon
release	David J. Kaufman David J. Kaufman
for	Charles A. Savant
date	6/16/81

DCEC PROTOCOLS STANDARDIZATION PROGRAM

TCP STANDARD: INITIAL VERSION

ABSTRACT

This document specifies the Transmission Control Protocol (TCP), a reliable connection-oriented transport protocol for use in packet-switched communication networks and internetworks. This document is a partial draft intended to focus attention on and stimulate discussion of essential TCP services. The document includes an overview with a model of operation, a description of services offered to users, and a description of services required of the network layer. A complete initial draft will be released in September, 1981.

CONTENTS

1. OVERVIEW

This document specifies the Transmission Control Protocol (TCP), a reliable connection-oriented transport protocol for use in packet-switched communication networks and interconnected sets of such networks. The document is organized into seven sections plus a bibliography and a glossary of terms. This, the first section, establishes TCP's role in the DoD protocol architecture and introduces TCP's major services and mechanisms. The second and third sections more formally specify the services TCP offers to upper layer protocols and the interface through which those services are accessed. Similarly, the next two sections specify the services required of the lower layer protocol and the lower interface. The sixth section specifies the mechanisms supporting the TCP services. The seventh section outlines the functionality required of the execution environment for successful TCP operation. The reader is assumed to be familiar with the DoD protocol architecture which defines a layered model of network communications systems[1].

This document incorporates the organization and specification techniques presented in the Protocol Specification Guidelines[2]. One goal of the guidelines is to avoid assuming a particular system configuration. As a practical matter, the distribution of protocol layers to specific hardware configurations will vary. For example, many computer systems are connected to networks via front-end computers which house TCP and lower layer protocol software. Although appearing to focus on TCP implementations which are co-resident with the upper and lower layer protocols, this specification can apply to any configuration given appropriate intra-layer protocols to bridge hardware boundaries.

TCP is designed to provide communication between pairs of processes in logically distinct hosts on networks and sets of interconnected networks. Thus, TCP serves as the basis for DoD-wide inter-process communication in communication systems. TCP will operate successfully in an environment where the loss, damage, duplication, or misordering of data, and network congestion can occur. This robustness in spite of unreliable communications media makes TCP well suited to support military, governmental, and commercial applications.

TCP appears in the DoD protocol hierarchy at the transport layer. Here, TCP provides connection-oriented data transfer that is reliable, ordered, full-duplex, and flow controlled. TCP is designed to support a wide range of upper layer protocols (ULPs) including those in the session layer, the presentation layer, and the application layer. The ULPs can channel continuous streams of data through TCP for delivery to peer ULPs. TCP breaks the streams into portions which are encapsulated together with appropriate addressing and control information to form a segment. A segment is the unit of exchange between peer TCPs. In turn, TCP passes the segments to the network layer for transmission through the network to the peer TCP.

```
+------+ +-----+ +------+         +-----+
|Telnet| | FTP | | EFTP |   ...   |     |
+------+ +-----+ +------+         +-----+
    |     |          |               |
  +-----+        +-----+          +-----+
  | TCP |        | UDP |    ...   |     |
  +-----+        +-----+          +-----+
     |              |                |
  +-----------------------------------+
  |         INTERNET PROTOCOL         |
  +-----------------------------------+
                    |
       +------------------------+
       |   Subnetwork Protocol  |
       +------------------------+
```

1. Example Host Protocol Hierarchy

The network layer provides for data transfer between hosts in a communication system. Such systems may range from a single network to interconnected sets of networks forming an internetwork. The minimum required data transfer service is limited; data may be lost, duplicated, misordered, or damaged in transit. As part the transfer service though, the network layer must provide global addressing, handle routing, and hide network-specific characteristics. As a result, upper layer protocols using the network layer may operate above a wide spectrum of subnetwork systems ranging from hard-wire connections to packet-switched or circuit-switched subnets. Additional services the network layer may provide include selectable levels of transmission quality such as precedence, reliability, transmission mode, and speed. The network layer also allows data labelling, needed in secure environments, to associate security and compartmentation information with data.

TCP is specifically designed to operate above the Internet Protocol (IP) which supports the interconnection of networks[3]. IP's internet datagram service and virtual network service provide the functionality described above. Originally, TCP and IP were developed as a single protocol providing resource sharing across different packet networks[4]. The need for other transport protocols to use IP's services led to their specification as two distinct protocols[3,5].

TCP builds its services on top of the network layer's potentially unreliable ones with mechanisms such as error detection, positive acknowledgments, sequence numbers, and flow control. These mechanism require certain addressing and control information to be initialized and maintained during data transfer. This collection of information is called a TCP connection. The following paragraphs describe the purpose and operation of the major TCP mechanisms.

TCP uses a positive acknowledgement with retransmission (PAR) mechanism to recover from the loss of a segment by the lower layers. The basic strategy with PAR is for a sending TCP to retransmit a segment at timed intervals until a positive acknowledgement is returned. The choice of retransmission timeouts impacts efficiency. Too long a timeout reduces data throughput while too short a timeout floods the transmission media with superfluous data. In TCP, the timeout is dynamically adjusted to approximate the segment round-trip time plus a fudge factor for internal processing.

TCP uses a simple checksum to detect segments damaged in transit. Such segments are discarded without being acknowledged. Hence, damaged segments are treated identically to lost segments and are compensated for by the PAR mechanism.

TCP assigns sequence numbers to identify each octet (an eight bit byte) of the data stream. These enable a receiving TCP to detect duplicate and out-of-order segments. Sequence numbers are also used to extend the PAR mechanism by allowing a single acknowledgment to cover many segments worth of data. Thus, a sending TCP can still send new data although previous data has not been acknowledged.

TCP's flow control mechanism enables a receiving TCP to govern the amount of data dispatched by a sending TCP. The mechanism is based on a "window" which defines a contiguous interval of acceptable sequence numbered data. As data is accepted, the window slides upward in the sequence number space. The flow control window is carried in every segment to enable peer TCPs maintain current window status.

TCP employs a multiplexing mechanism to allow many ULPs within a single host to use TCP simultaneously. This mechanism associates identifiers, called ports, to ULPs accessing TCP services to A ULP is uniquely identified with a socket, the concatenation of a port and an internet address. Each communicating ULP-pair, or connection, is uniquely named with a socket pair. This naming scheme allows a single ULP to support connections to multiple remote ULPs. ULPs which provide popular resources are assigned permanent sockets, called well-known sockets. Some well-known sockets currently used in DoD networks are published in in [6] and [7].

When two ULPs wish to communicate, they instruct their TCP's to initialize and synchronize the mechanism information on each side to open the connection. However, the potentially unreliable network layer can complicate the process of synchronization. Delayed or duplicate segments from previous connection attempts might be mistaken for new ones. A handshake procedure with clock based sequence numbers is used in connection opening to reduce the possibility of such false connections. In the simplest handshake, the TCP pair synchronizes sequence numbers by exchanging three segments, thus the name three-way handshake. The scenario following the overview depicts this exchange. The procedure will be discussed more fully in the mechanism section. A comprehensive study of connection establishment and related connection management issues can be found in [6].

A ULP can open a connection in one of two modes, passive or active. With a passive open, a ULP instructs its TCP to be "receptive" to connections with other ULPs. With an active open a ULP instructs its TCP to actively initiate a three-way handshake to connect to another ULP. Usually, an active open is targeted to a passive open. This active/passive model supports server-oriented applications where a permanent resource, such as a data-base management process, can always be accessed by remote users. However, the three-way handshake also coordinates two simultaneous active opens to establish a connection.

Over an open connection, the ULP-pair can exchange a continuous stream of data in both directions. Normally, TCP transparently groups the data into TCP segments for transmission at its own convenience. However, a ULP can exercise an "end-of-letter" (also called EOL) mechanism to force TCP to package and send data passed up to that point without waiting for additional data. This mechanism is intended to prevent possible deadlock situations where a ULP waits for data internally buffered by TCP. For example, an interactive editor might wait forever for a single input line from a terminal. An EOL pushes data through the TCPs to the awaiting process. TCP also provides a means for a sending ULP to signal a receiving ULP that "urgent" data appears in the upcoming data stream. This signalling mechanism is intended for use as an interrupt or break.

When data exchange is complete the connection can be closed by either ULP to free TCP resources for other connections. Connection closing can happen in two ways. The first, called a graceful close, is based on the three-way handshake procedure to complete data exchange and coordinate closure between the TCPs. The second, called an abort, does not allow coordination and may result in loss of unacknowledged data.

1.1 Scenario

The following scenario provides a walk-through of a connection opening, data exchange, and a connection closing as might occur between the data base management process and user mentioned above. The scenario glosses over many details to focus on the three-way handshake mechanism in connection opening and closing, and the positive acknowledgement with retransmission mechanism supporting reliable data transfer. Although not pictured, the network layer transfers the information between the TCPs. For the purposes of this scenario, the network layer is assumed not to damage, lose, duplicate, or re-order data unless explicitly noted.

The scenario is organized into three parts:

a. A simple connection opening in steps 1-7.

b. Two-way data transfer in steps 8-18.

c. A graceful connection close in steps 19-27.

The following notation is used in the diagrams:

```
 < < < SEQ# 200 < < <      depicts information exchange
 > > > ACK# 201 > > >      between peer TCPs

     :          ^          depicts information passing
 SEND DATA      :          across the interface between
     :     DELIVER DATA    a ULP and its TCP
     v          :
```

```
                ----------                                     ----------
                | ULP A  |                                     | ULP B  |
                ----:-----                                     ----:-----
                    :                                              :
              2.ACTIVE OPEN                                  1.PASSIVE OPEN
                  TO B                                             :
                    :                                              :
               -----v----                                     ----v-----
               |         |> > > > 3.SYN SEQ#200> > > > > >|         |
               | TCP A   |                                  | TCP B   |
               |         |                                  |         |
               ----------                                     ----------
```

1. ULP B (the DB manager) issues a PASSIVE OPEN to TCP B to prepare for connection attempts from other ULPs in the internetwork.

2. ULP A (the user) issues an ACTIVE OPEN to open a connection to ULP B.

3. TCP A sends a segment to TCP B with an OPEN control flag, called a SYN, carrying the first sequence number (shown as SEQ#200) it will use on its half, the "A" half, of the full-duplex connection.

```
           ----------                                          ----------
           | ULP A  |                                          | ULP B  |
           ---^------                                          ----^-----
              :                                                    :
      6.CONNECTION OPEN                                   7.CONNECTION OPEN
             TO B                                                TO A
              :                                                    :
           ----:-----                                          ----:-----
           |        |< < < <4.ACK#201;SYN SEQ#550< < < <|          |
           | TCP A  |                                    | TCP B    |
           |        |> > > > > >5.ACK#551> > > > > > > >|          |
           ----------                                          ----------
```

4. TCP B responds to the SYN by sending a positive acknowledgement, or ACK, marked with TCP A's next sequence number. In the same segment, TCP B sends its own SYN with the first sequence number for the "B" half of the full-duplex connection (SEQ#550).

5. TCP A responds to TCP B's SYN with an ACK showing the next sequence number expected from B.

6. TCP A now informs ULP A that a connection is open to ULP B.

7. Upon receiving the ACK, TCP B may inform ULP B that a connection has been opened to ULP A.

```
    ---------                                          ---------
    | ULP A |                                          | ULP B |
    ---:-----                                          ----^----
       :                                                   :
  8.SEND DATA                                     10.DELIVER DATA
       :                                                   :
   ----v-----                                          ----:-----
   |        |> > > > >9.DATA SEQ#201 > > > > > >|          |
   | TCP A  |                                   |  TCP B   |
   |        |< < < < 11. ACK SEQ#221 < < < < < <|          |
   ----------                                          ----------
```

8. ULP A passes 20 octets of data to TCP A for transfer across the open connection to ULP B.

9. TCP A packages the data in a segment marked with current "A" sequence number.

10. After validating the sequence number, TCP B accepts the data and delivers it to ULP B.

11. TCP B sends an ACK of the accepted data marked with the sequence number of the next data octet expected.

```
   ---------                                           ---------
   | ULP A |                                           | ULP B |
   ----^----                                           ----:----
       :                                                   :
  14.DELIVER DATA                                    12.SEND DATA
       :                                                   :
   ----:-----                                          ----v-----
   |        |< < < < 13.DATA SEQ#551 < < < < < <|          |
   |        |                                   |          |
   |        |> > > > 15. ACK #726               |          |
   | TCP A  |                                   |  TCP B   |
   |        |< < < < 16.DATA SEQ#551 < < < < < <|          |
   |        |                                   |          |
   |        |> > > > 17.ACK #726 > > > > > > > >|          |
   ----------                                          ----------
```

12. ULP B passes 125 octets of data to TCP B for transfer to ULP A.

13. TCP B packages the data in a segment marked with the "B" sequence number.

14. TCP A accepts the segment and delivers the data to ULP A.

15. TCP A returns an ACK of the received data marked with the sequence number of the next expected data octet. However, the segment is lost by the network and never arrives at TCP B.

16. TCP B times out while waiting for the ACK and retransmits the segment. TCP A receives the retransmitted segment, but discards it because the data from the original segment has already been accepted. However, TCP A re-sends the ACK.

17. TCP B gets the second ACK.

```
         ---------                                           ---------
         | ULP A |                                           | ULP B |
         ---:-----                                           ----^----
            :                                                    :
     18.CLOSE CONNECTION                                 20.ULP A CLOSING
          TO B                                                   :
            :                                                    :
        ----v-----                                           ----:-----
        |        |> > > > 19. FIN SEQ#221 > > > > > >|          |
        | TCP A  |                                   |  TCP B   |
        |        |< < < < 11. ACK#222 < < < < < < < <|          |
        ----------                                           ----------
```

18. ULP A closes its half of the connection by issuing a CLOSE to TCP A.

19. TCP A sends a segment marked with a CLOSE control flag, called a FIN, to inform TCP B that ULP A will send no more data.

20. TCP B gets the FIN and informs ULP B that ULP A is closing.

21. TCP B returns an ACK of the FIN.

```
        ---------                                            ---------
        | ULP A |                                            | ULP B |
        ---^-----                                            ----:----
           :                                                     :
      25.CONNECTION                                     22.CLOSE CONNECTION
         CLOSED                                                 TO A
           :                                                     :
        ---:------                                           ----v-----
        |        |< < < < 23. FIN SEQ#576 < < < < < <|          |
        | TCP A  |                                   |  TCP B   |
        |        |> > > > 24. ACK#577 > > > > > > > >|          |
        ----------                                           ----------
```

22. ULP B completes its data transfer and closes its half of the connection.

23. TCP B sends a FIN to TCP A to show ULP B's closing.

24. TCP A gets the FIN and responds with a ACK to TCP B.

25. TCP A informs ULP A that the connection is closed.

26. (Not pictured) TCP B receives the FIN ACK from TCP A and informs ULP B that the connection is closed.

2. UPPER LEVEL SERVICE DESCRIPTION

This section describes the services offered by the Transmission Control Protocol to upper layer protocols (ULPs). The goals of this section are to provide the motivation for protocol mechanisms and to provide ULPs with a definition of the functions provided by this protocol.

The services provided by TCP can be organized as follows:

- o multi-user support service
- o connection management services
- o data transport service
- o error reporting service

A description of each service follows.

2.1 Multi-user Support Service

TCP shall provide services to multiple pairs of upper layer protocols within the internetwork environment. A ULP using TCP services shall be identified with a "port". A port, when concatenated with an internet address, forms a socket which uniquely names a ULP throughout the internet. TCP shall use the pair of sockets corresponding to a ULP-pair to differentiate between multiple users.

2.2 Connection Management Service

TCP shall provide data transfer capabilities, called connections, between pairs of upper layer protocols. A connection provides a private communication channel between two ULPs. Characteristics of data transfer are specified in the data transfer service description. Connection management can be broken into three phases: connection establishment, connection maintenance, and connection termination.

2.2.1 Connection Establishment

TCP shall provide a means to open connections between ULP-pairs. Connections are endowed with certain properties that apply for the lifetime of the connection. These properties, including security and precedence levels, are specified by the ULPs at connection opening. Connections can be opened in one of two modes: active or passive.

TCP shall provide a means for a ULP to actively initiate a connection to another ULP uniquely named with a socket. TCP shall establish a connection to the named ULP if:

1. no connection between the two named ULPs already exists,

2. internal TCP resources are sufficient,

3. the other ULP exists, and has

   a. simultaneously executed a matching active open to this ULP,

   b. or, previously executed a matching passive open.

TCP shall provide a means for a ULP to listen for and respond to active opens from correspondent ULPs. Correspondent ULPs are named in one of two ways:

1. fully specified : A ULP is uniquely named by a socket. A connection is established when a matching active open is executed (as described above) by the named ULP.

2. unspecified : No socket is provided. A connection is established with any ULP executing an matching active open naming this ULP.

2.2.2 Connection Maintenance

TCP shall maintain established connections supporting the data transfer service described in Section 2.3. And, TCP shall provide a means for a ULP to acquire current connection status with regard to connection name, data transfer progress, and connection qualities.

2.2.3 Connection Termination

TCP shall provide a means to terminate established connections and nullify connection attempts. Established connections can be terminated in two ways:

1. Graceful Close : Both ULPs close their side of the duplex connection, either simultaneously or sequentially, when data transfer is complete. TCP shall coordinate connection termination and prevent loss of data in transit as promised by the data transfer service.

2. Abort : One ULP independently forces closure of the connection. TCP shall not coordinate connection termination. Any data in transit may be lost.

TCP shall allow a ULP to nullify either passive and active opens any time before connection opening completes.

2.3 Data Transport Service

TCP shall provide data transport over established connections between ULP-pairs. The data transport is full-duplex, error-free, guaranteed, ordered, labelled with security and precedence levels, and flow controlled. A more detailed description of each of the data transport characteristics follows.

- full-duplex : TCP shall support simultaneous bi-directional data flow between the correspondent ULPs.

- error free : TCP shall deliver data that is free of errors within the probabilities supported by a simple checksum.

- guaranteed : TCP shall promise data delivery within a ULP specified time limit barring host, network, or internetwork failure. When system conditions prevent timely delivery, TCP shall notify the ULP-pair of service failure and subsequently terminate the connection.

- ordered : TCP shall deliver data to a destination ULP in the same sequence as it was provided by the source ULP.

- labelled : TCP shall associate with each connection with the security and precedence levels supplied by the ULPs during connection establishment. When this information is not provided by the ULP-pair, TCP shall assume default levels. TCP shall establish a connection between a ULP-pair only if the security/compartment information exactly matches. If the precedence levels do not match, the higher precedence level is associated with the connection.

- flow controlled : TCP shall regulate the flow of data across the connection to prevent internal TCP congestion leading to service degradation and failure.

TCP shall provide two capabilities to ULPs concerning data transfer over an established connection: data stream partitioning and urgent data signalling.

- data stream partitioning : TCP shall allow a sending ULP to mark places in the data stream as "end-of-letter". These markings signify that the data should be delivered to the correspondent ULP without delay. TCP shall transmit data preceding and including the end-of-letter mark to the receiving TCP without waiting for additional data. The receiving TCP shall deliver the data to the receiving ULP in the same manner.

- urgent data signalling : TCP shall provide a means for a sending ULP to inform a receiving ULP of the presence and location of significant data in the upcoming data stream.

2.4 Error Reporting Service

TCP shall report service failure stemming from catastrophic conditions in the internetwork environment for which TCP cannot compensate. Such catastrophic conditions include host failure, internetwork partitioning, subnetwork failure, and internetwork failure.

3. UPPER LAYER SERVICE/INTERFACE SPECIFICATION

Although omitted from this draft, this section is intended to specify the TCP services provided to upper layer protocols and the interface through which these services are accessed. The first part will define the interaction primitives and their interface parameters. The second part will contain the abstract machine specification of the services offered to the upper layer.

4. DESCRIPTION OF LOWER LAYER SERVICE REQUIREMENTS

This section describes the minimum set of services required of the network layer protocol by TCP.
The services required are:

- o data transfer service
- o virtual network service
- o error reporting service

A description of each service follows.

4.1 Data Transfer Service

The network layer protocol must provide data transfer between TCP modules in a communication system. Such a system may consist of a single network or a set of interconnected networks forming an internet. Data must arrive at a destination with non-zero probability; some data loss may occur. The data transfer service is not required to preserve the order in which portions of data are supplied by the source upon delivery at the destination. Data delivered is not necessarily error-free.

The network layer protocol must provide transparent data transfer throughout the system. TCP need only supply global addressing and control information with each portion of data to be delivered. Routing and network specific characteristics are handled by the network layer protocol. For example, TCP need not be aware of current topology or packet size restrictions to transmit segments through a network.

4.2 Virtual Network Service

The network layer protocol must provide a means for TCP to select from the transmission service qualities provided by the communication system for each portion of data delivered. The transmission quality parameters must include precedence. Also, the network layer protocol must provide a means of labelling each portion of data with security level and compartmentation information.

4.3 Error Reporting Service

The network layer protocol must provide error reports to TCP indicating discontinuation of the above services caused by catastrophic conditions in this or lower layer protocols.

5. LOWER LAYER SERVICE/INTERFACE SPECIFICATION

Although omitted from this draft, this section is intended to specify the minimal network layer services required by TCP and the interface through which those services are accessed. The first part of this section will define the interaction primitives and their parameters for the lower interface. The second part will contain the abstract machine specification of the lower layer services and interaction discipline.

6. MECHANISM SPECIFICATION

Although omitted from this draft, this section is intended to define the mechanisms supporting the services offered by TCP. The first subsection will motivate the specific mechanisms chosen and discuss the underlying philosophy of those choices. The second subsection will define the format and use of the TCP header fields. The last subsection will specify the local protocol operation with an extended state machine model.

7. EXECUTION ENVIRONMENT REQUIREMENTS

Although omitted from this draft, this section is intended to describe the services required from the execution environment for the proper implementation and operation of TCP.

8. GLOSSARY

Acknowledgement Number
A 32-bit field of the TCP header containing the next sequence number expected by the sender of the segment.

ACK
Acknowledgement flag: a control bit in the TCP header indicating that the acknowledgement number field is significant for this segment.

Checksum
A 16-bit field of the TCP header carrying the one's complement based checksum of both the header and data in the segment.

datagram
A self-contained package of data carrying enough information to be routed from source to destination without reliance on earlier exchanges between source or destination and the transporting network.

datagram service
A datagram, defined above, delivered in such a way that the receiver can determine the boundaries of the datagram as it was entered by the source. A datagram is delivered with high probability to the desired destination, but it may possibly be lost. The sequence in which datagrams are entered into the network by a source is not necessarily preserved upon delivery at the destination.

Data Offset
A TCP header field containing the number of 32-bit words in the TCP header.

Destination Port
The TCP header field containing a 2-octet value identifying the destination of a segment's data.

EOL
A control bit of the TCP header indicating that a segment contains a data boundary.

FIN
A control bit of the TCP header indicating that no more data will be sent by the sender.

gateway
A device, or pair of devices, which interconnect two or more data networks enabling the passage of data from one network to another. In this architecture, a gateway contains at least an IP module and a subnetwork protocol(SNP) for each connected network.

header
Collection of control information transmitted with data between peer entities.

host

A computer, particularly a source or destination of messages from the point of view of the communication network.

IMP
Interface Message Processor: the packet switch of the ARPANET.

internetwork
A set of interconnected subnetworks.

internet address
A four octet (32 bit) source or destination address composed of a Network field and a Local Address field.

internet datagram
The package exchanged between a pair of IP modules. It is made up of an internet header and a data portion.

local address
The address of a host within a network. The actual mapping of an internet local address onto the host addresses is quite general, allowing for many to one mappings.

local network
The network directly attached to host or gateway.

module
An implementation, usually in software, of a protocol or other procedure.

octet
An eight bit byte.

Options
The optional set of fields at the end of the TCP header used to carry buffer size information.

packet
The unit of data transmitted by a packet-switched network. A packet usually contains nested control information and data from the higher layer protocols using the network.

packet network
A network based on packet-switching technology. Messages are split into small units (packets) to be routed independently on a store and forward basis. This packetizing pipelines packet transmission to effectively use circuit bandwidth.

Padding
A header field inserted after option fields to ensure that the data portion begins on a 32-bit word boundary. The padding field value is zero.

protocol
A discipline between peer components for the exchange of data.

Precedence
One of the service quality parameters in the Internet Protocol's type of service mechanism. Precedence is a measure of datagram importance. This parameter can be set to routine, priority, immediate, flash, or flash override.

reliability
A quality of data transmission defined as guaranteed, ordered delivery.

Reserved
A 6-bit field of the TCP header that is not currently used but must be zero.

RST
A control bit of the TCP header indicating that the connection associated with this segment is to be reset.

segment
The unit of data exchanged by TCP modules. This term may also be used to describe the unit of exchange between any transport protocol modules.

Sequence Number
A 32-bit field of the TCP header containing the sequence number of the first byte of data carried in the segment.

Source Port
The TCP header field containing a 2-octet value identifying the source of a segment's data.

stream delivery service
The special handling required for a class of volatile periodic traffic typified by voice. The class requires the maximum acceptable delay to be only slightly larger than the minimum propagation time, or requires the allowable variance in packet interarrival time to be small.

TCP segment
The unit of data exchanged between TCP modules (including the TCP header).

ULP
Upper Level Protocol: any protocol above TCP in the layered protocol hierarchy that uses TCP. This term includes presentation layer protocols, session layer protocols, and user applications.

Urgent Pointer
A TCP header field containing a a positive offset to the sequence number of the segment indicating the position of urgent data in the connection's data stream. This field is valid only when the URG flag is on.

URG
A control bit of the TCP header indicating that the urgent field contains a valid pointer to urgent information in the connection's data stream.

user
A generic term identifying a process or person employing the Transmission

Control Protocol. This term may describe a session layer protocol, a presentation layer protocol, or an application program.

Window
A 2-octet field of the TCP header indicating the number of data octets (relative to the acknowledgement number in the header) that the segment sender is currently willing to accept.

9. BIBLIOGRAPHY

1. "DCEC Protocols Standardization Program Preliminary Architecture Report" TM-7038/200/00, Contract No. DCA100-80-C-0044, February 1981.

2. "Protocol Specification Guidelines", TM-7038/204/00, Contract No. DCA100-80-C-0044, June 1981.

3. J. Postel (ed.), "DoD Standard Internet Protocol", Defense Advanced Research Projects Agency, Information Processing Techniques Office, RFC760, IEN128, January 1980.

4. V. Cerf and R. Kahn, "A Protocol for Packet Network Interconnection", IEEE Transactions on Communications, May 1974.

5. J. Postel (ed.), "DoD Standard Transmission Control Protocol", Defense Advanced Research Projects Agency, Information Processing Techniques Office, RFC761, IEN129, January 1980.

6. C. Sunshine and Y.Dalal, "Connection Management in Transport Protocols", Computer Networks vol.2, 1978, pp.454-473.

7. J. Postel, "Address Mappings", IEN 115, USC/Information Sciences Institute, August 1979.

8. J. Postel, "Assigned Numbers", RFC 762, IEN 127, USC/Information Sciences Institute, January 1980.

14919

NETWORK WORKING GROUP
RFC #475
NIC #14919

ABHAY K. BHUSHAN
MIT-DMCG
MARCH 6, 1973

FTP AND NETWORK MAIL SYSTEM

This paper describes my understanding of the results of the Network Mail System meeting SRI-ARC on February 23, 1973, and the implications for FTP (File Transfer Protocol). There was general agreement at the meeting that network mail function should be within FTP.

FTP currently provides two commands for handling mail. The MAIL command allows a user to send mail via the TELNET conection (the server collects the mail and determines its end by searching for the character sequence "CRLF.CRLF"). The MLFL (mail file) command allows a user to send mail via the data connection (requires a user-FTP to handle the command but transfer is more efficient as server need not search for a special character sequene). These commands are being used to provide network mailing facilities. Local mail and SNDMSG programs have been modified at many sites to include network mailing (e.g, USER@HOST at BBN-TENEX and MAIL host user at MIT-DMCG).

The network mail system should provide a facility whereby users can conveniently send messages to other network users who have "mailboxes" at one or more hosts. It is not required that the messages or mail be delivered in real-time. The network mail system is not an interactive inter-console communication facility, but it may be possible for some sites to deliver "urgent" mail to users in real-time (e.g., print mail at user console if user is currently logged-in). The mail system also does not provide a general inter-process communication facility, though it may be possible to deliver messages to programs which have mailbox addresses. Inter-process and inter-entity communication facilities are very desirable but are beyond the scope of the network mail system.

The concepts of "mailbox" and "mailbox addresses" are central to this discussion of network mail system. A mailbox is a place where the the mail is stored before a user picks it up. It may be a file in the user's directory or it may be a bin for hard-copy. The mailbox address is the address required by the sender in order to send the mail to its destination mailbox. For users who have an "on-line" network mailbox, the mailbox address contains the Host address and the user's mailbox identification at that Host. The mailbox identification is that which is required by an FTP-server in order that it may put the mail in the desired mailbox. The terms mailbox address and address will be used to refer to the on-line network mailbox address.

NETWORK MAIL SYSTEM FUNCTIONS

The network mail system should provide the following six functions:

1. CREATING: This refers to the manner in which the user creates or composes his message. The FTP servers do not explicitly provide any message editing capability (server's editing conventions may be applicable in the case of MAIL command). Editing conventions such as those for character delete and line cancel vary widely over the network. The user is most familiar with his local Host conventions and these should be used for network mail editing. The user also has access to local editing systems which can be used for composing message files. The message file may then be transmitted via the MAIL or MLFL commands (MLFL being preferable). The present FTP approach of assuming the creation of messages to be sender's responsibility seems adequate. TIP users if they desire editing facilities should use intermediate Hosts for creating and sending messages.

2. LOCATING: How sender determines receivers address. FTP assumes that the sender knows the receivers correct address. There is no published or "on-line" list of mailbox addresses. There is, however, a list of network participants maintained (on-line) and published by the Network Information Center (NIC) at SRI-ARC. The network users have been assigned a unique "NIC Ident" and Host site by the NIC. It was therefore specified in FTP that FTP-servers maintain a table that maps NIC Idents to mail-box identifications. The NIC will maintain on-line and publish the local mailbox address information for network participants. It would be possible for users to look up a published list, or querry the NIC on-line to locate destination addresses. The NIC will also provide an on-line facility (similar to FTP) that can be used by programs for retrieving the address information. This latter approach of the NIC's maintaining addresses has several advantages. The user can obtain a number of addresses for a group, and use these to transmit mail. The FTP servers need not maintain NIC Ident Tables, and the NIC can provide a good facility for locating addresses from last names, NIC idents, or even sketchy information. It may still be desirable that FTP servers accept NIC idents, last names, and other standard forms as mailbox identifiers.

3. SENDING: How message is sent to the destination mailbox. The messages may be sent directly to the destination mailbox (via TELNET or Data connections) or via an intermediate Host such as the NIC. FTP does not explicitly provide for mail forwarding by intermediate Hosts but FTP servers may be able to recognize addresses as not being local, and forward mail. In the event mail is to be forwarded, a desirable facility is to have the intermediate site return an acknowledgment (by request) upon delivery of mail or if delivery fails within a specified time. The current FTP specifications recommend that FTP-servers accept multiple addresses but do not require this.

4. STORING: Where mail is stored before reading and if information is available for later reference or retrieval. The FTP does not require that sender store mail or keep duplicate copies. It is the receiver's responsibility to store the information for reading, reference, or retrieval. The receiver need not store the mail as a data file but can directly print it out on a user console or line printer. FTP does not specify the procedures for storage handling by intermediate sites. If intermediate site is used for forwarding the mail then it should be the responsibility of that site to store mail until it is delivered to its final destination. If the mail is undeliverable then the intermediate site should return the undelivered information to the sender. A similar situation arises when sending of mail is deferred by the sending site (destination host may be down). The sending site then acts as an intermediate forwarder insofar as the user is concerned.

5. RECORDING: Should the mail be catalogued and recorded for later reference and retrieval. FTP currently does not provide an explicit mechanism for the receiver to record mail. If an intermediate site (the NIC) is used for mail distribution then a function of such a site could be to record mail, if so requested. NIC is ideal for recording mail, but other sites may also wish to record mail. If the mail is recorded, then it is not necessary to send the entire contents of the mail. Instead only a citation for the document can be sent and the receiver can retrieve the mail only if he wants to. This is particularly useful for large documents such as NWG/RFC which are distributed to a group. The citation may contain author, title, retrieval pathname, and perhaps an abstract.

6. READING: How the mail is finally presented to and read by the user. FTP currently assumes that mail reading is entirely the receiving site's function. However, there are ways in which the sender can aid the receiver in providing improved mail reading facilities. For example, the receiving system, if it knows a message to be urgent can deliver it immediately at a user console. Long messages may be put in separate files with notification in user's regular mail. Alternately, mail could be a citation that the reading program can retrieve upon user request. Selective handling of different classes of mail is important for an improved network mail system.

MODELS FOR MAIL SYSTEM USE

The user of a mail system can use intermediate site for locating addresses, recording and/or distributing mail, and for creating and reading mail. We therefore have the following models for mail system use:

1. The user connects directly to the destination FTP server and sends mail using the MAIL command. Local editing functions are limited to character delete and line cancel (assuming user is in line-a-time mode) and server conventions may also apply. The user only needs a user-TELNET program at his site but needs to know the destination address. This model is specially applicable to TIP and other mini-Host users who do not have a user-FTP or user-Mail programs.

2. The user composes the mail using a local editor (or mail system) and then requests his user-FTP or mail program to send the mail directly to the destination via the FTP MAIL or MLFL commands. The user needs to know the destination address. The mail can be deferred by the sending program if the destination Host is down. TIP users can use this model by using the facilities of a "home-base" Host.

3. The user uses an intermediate site such as NIC (other sites may provide forwarding services too) for mail distribution. The user need not know the destination addresses but can use NIC idents for individuals and groups of individuals. The mail can be recorded on request and its sending can be deferred (the destination Host may be down, or it may be more economical to defer mail). The message to be mailed may be created at the local site using local editing facilities, or it may be created directly at the intermediate site.

4. The user may send a citation of the mail instead of the complete mail item. The citation refers to an existing document which can be retrieved on-line (such as the NIC number of a NIC journal communication).

MAILING TO TIP USERS

The TIP does not currently provide an FTP server or mailbox facilities. While it is possible to send mail to TIP terminals (such as line printers) it seems undesirable to do so because of the possibility of losing mail, the lack of privacy, and the fact that user may be several (or several hundred) miles away from the location of the TIP. The TIP users normally have a "home-base" computer where they do their computing work most of the time. The TIP user problem is best solved by requiring that TIP users rent mailboxes at their "home-base" Host. Such a Host can provide good mail reading and querry facilities. A TIP user can request his "home" Host to send him notification of mail on a TIP terminal. If RDML command (NWG/RFC 458) is accepted in FTP, TIP users could use such a command. More important, if the user has a number of mailboxes on different Hosts, the RDML (or RDMF) command can be used to read his mail at all the sites where he has mailboxes.

ACCESS CONTROL IN MAIL SYSTEM

It has been suggested that FTP specification should require that mail function (for receiving mail) should be "free", i.e., FTP servers should not require the user to "login" (send the USER, PASS, and ACCT commands). In the absence of the access control commands the FTP server should charge the cost of receiving mail to an overhead or browsing account. It should be noted that this "free" mail function using default "USER" account may not allow non mail-related commands without reinitializing. This requirement will improve communication among the network users.

Some systems, such as Multics, have mechanisms for access control in the receipt of mail. That is a user can specify who is eligible to send him mail (normally users give the access "*.*.*", i.e., any one can send mail). The access control commands would be required to gain privileged access. The USER command does not seem the best way to identify the sender of mail. Consider users logged in as GUEST, ICCC, NETWORK, MIT-DMCG, and NETWORK-USER. A separate FROM command seems desirable. Such a ommand can be used to identify the sender as well as to send acknowledgments and replies. The receiving site can tag the mail as: FROM AKB at MIT-DMCG, logged in as GUEST. The receiver can then send reply to the mailbox address AKB at Host 70 (SNDMSG AKB@DMCG or MAIL DMCG AKB).

NETWORK INFORMATION CENTER FUNCTIONS

The NIC is a very special facility for handling mail. It provides facilities for recording and distributing mail to individuals and groups of individuals, and for locating users' addresses. The NIC will also undertake to provide distribution of unrecorded mail. Currently the NIC requires that users log into the NIC and use NLS to create and distribute mail. Using NLS for creating mail has been a frustrating experience for many who are used to different editing systems. Recently there has been a problem that NIC is overloaded at most times of the day and even if one can get a "network terminal" and log in, the interaction is quite slow. As NIC (or NLS) is designed for character-at-a-time interaction with remote echo, the use is inefficient. Using NIC is particularly unbearable when the user falls behind in his echo by as much as an entire line.

An alternative to direct use of NIC is to use the NIC via FTP and programs at the user's site. The user can create journal documents using his own local editing system and then transfer it to NIC via FTP. The user may have to specify such information as author, title, where the acknowledgment should be sent, and journal number if the item is to be recorded. It should also be possible for users to send sequential files to NIC and have them restructured into NLS form without having to do an "input sequential" (a suggestion is to "NLS" the file if its name is suffixed with a .NLS). Alternately it should be possible for user's to retrieve journal documents and other sequential files without having to do a previous "output sequential".

The NIC curently delivers mail via hardcopy and/or on-line. On-line currently means that user must log into NIC to see if he has a message and read it by "print branch". The messages are not seen by the destination users for several days and many users get their hard copy before they have had a chance to examine their on-line NIC mail. If the NIC were to deliver mail via FTP to network users, then the mail turn-around time will be greatly speeded and the users will not have to log into the NIC. Large documents need not be mailed to the user in their entirety but only a citation need be sent. The NIC will have to collect the information on the mailbox addresses of Network participants for delivering mail, especially since it appears that many FTP servers are not "respecting" NIC idents. It is recognized that a user may have more than one valid mailbox address, but the NIC needs to have only one (the most used) of these addresses.

The NIC identification subsystem (currently accessable via NLS only) contains information on users (such as affiliation, US Mail address, telephone numbers, etc.) and groups (members, etc.). The on-line mailbox address information can be added here. The NIC will undertake to provide a facility whereby the identification subsystem can be querried by programs, allowing mailing programs to retrieve the addresses automatically. This facility will be separate from FTP.

FTP MODIFICATIONS

The FTP currently does not provide explicit facilitiies for recording mail, communicating sender's address, sending program readable citations, specifying author and title for documents, requesting acknowledgments, and indicating message type (urgent, ordinary, and long). To overcome these deficiencies, we can take any of the following approaches:

1. Kludge the desired features in the pathname syntax of the MAIL and MLFL commands, justifying the kludge on the grounds that most of the functions are to be used only by the NIC.

2. Add new commands for the desired functions and alter the MAIL and MLFL commands somewhat to recognize the existence of the new commands.

3. Define a new mail command which incorporates the missing functions (in the process defining new commands for the desired functions). The MAIL and MLFL commands can be used in their

present form but may be gradually phased out.

The first approach seems undesirable to me as many of the missing functions can be used by other sites as well. In addition it will be easier to write programs to deal with commands rather than a complex syntax. The second and the third approaches are not very different from each other. The third approach seems preferable as it will allow existing mail programs to function in their present form. Using the third approach consider the following new FTP commands:

1. MLTO (mail to): The argument is one or more mailbox identifiers separated by "," (commas). It is suggested that if there is no argument, the mail should be sent to some responsible user or printed on a printer. This command starts the sequence of optional FTP mail related commands described below. The sequence ends with the TEXT, FILE, or CITA (citation) commands.

2. FROM: The argument is the address of the sender or senders. It is in a standard form that can be interpreted by programs as well as human users. The information is to be used for identifying the sender(s), for sending replies, and for sending acknowledgments if the receiver is an intermediate forwarding site.

3. MTYP (mail type): This identifies the type of mail as U (urgent), O (ordinary), and L (long). The receiving system can take the appropriate actions from this knowledge. The default assumption is ordinary mail.

4. RECO (record the mail): The argument if present is the identifying information for recording (such as NIC Journal number). If no argument is present the server will assign the recording information and send an appropriate reply (real-time or deferred).

5. AUTH (author): Identifies the author of the document in a form acceptable to the server (NIC ident may be required by NIC).

6. TITL (title): Identifies the title of the document. The argument is an ASCII string ending with the sequence "CRLF.CRLF".

7. ACKN (acknowledge): Relevant for intermediate forwarding sites. Asks the server to send acknowledgment on delivery or if delivery fails within a specified time.

8. TEXT: No arguments. Starts the transfer of mail over TELNET connection in an identical manner as MAIL.

9. FILE: No arguments. Starts transfer of mail over the data connection in an identical manner as MLFL.

10. CITA (citation): Argument is the pathname of a retrievable file.

We also need to define new reply codes for handling mail. Some sites have expressed the need for replies such as "send only X bytes of mail". Other replies could specifically request additional commands such as USER/PASS/ACCT for priviliged mailing, FROM/ACKN for mail forwarding, and AUTH/TITL for recorded mail. Another suggestion that may be given consideration is allowing TYPE/BYTE other than A/8 for FILE command. Mailing large files between like machines such as PDP-10s is more efficient in I/36. The RDML and RDMF commands proposed by Bressler and Thomas (NWG/RFC 458) also merit consideration as they would aid the handling of mail for users who have mailboxes at different Hosts.

NETWORK WORKING GROUP
Request for Comments #278
NIC 8056
Catories: A.5, 0.7
Obsoletes RFC 221, NIC 7612

Abhay Bhushan, MIT-DMCG 1
Bob Braden, UCLA-CCN 2
Eric Harslem, RAND 3
John Heafner, RAND 4
Alex McKenzie, BBN-NET 5
John Melvin, SRI-ARC 6
Bob Sundberg, HARV 7
Dick Watson, SRI-ARC 8
Jim White, UCSB 9
17-Nov-1971 10

REVISION OF THE MAIL BOX PROTOCOL 11

The file transfer committee met and discussed the Mail Box Protocol RFC 221, NIC 7612. The potential utility for the mechanism was confirmed and a couple of changes suggested. We first give the changes and then restate the Protocol. 12

CHANGES 13

1) The Mail Box Protocol is only to allow ASCII strings of text formatted for a network standard line printer rather than allowing other data types. 13a

2) A new command is to be added to the File Transfer Protocol called "Append with Create" which appends to a file if the file exists, and creates a file if it does not exist. 13b

3) The standard path name for the mailbox is to be, using conventional metalanguage symbols, 13c

"MAIL" <separator> ("PRINTER"/<ident>) 13c1

<separator> is the ASCII GS, octal 035. The semantics of the above are the following: 13c2

<ident> is a NIC IDENT. 13c3

"MAIL" <separator> "PRINTER" would be interpreted by the receiving site as meaning Append with Create the transmitted file to a bulk mail file to be printed or directly output it to a printer. 13c4

"MAIL" < separator> <ident> would be interpreted to mean either 13c5

1) The same as "MAIL" <separator> "PRINTER" i.e., ignore <ident> or 13c5a

2) Append with Create the following file to a file specifically for the person designated by <ident> for either online access or printing or both. 13c5b

The problem of delivering mail to TIPs was discussed. 13d

At the moment TIPs support only the Telnet Protocol, but it is planned to support the Data Transfer Protocol. TIPs will have an ASCII line printer available as an optional device. People desiring to send a mail item to a TIP with a Printer can open a standard published socket and transmit to it with Telnet Protocol now, later also with the Data Transfer Protocol. The NIC's plans with regard to TIPs is not to do automatic network delivery to them. Messages to people using TIPs can be sent to them through the NIC and will be delivered as with everyone else directly to the person's initial file at the NIC. The TIP user can read the item online or obtain a hardcopy at his terminal with the Output Device Teletype command of NLS. 13e

MAIL BOX PROTOCOL 14

The Mail Box Protocol will use established network conventions, specifically the Network Control Program, Initial Connection Protocol, Data Transfer Protocol, and File Transfer Protocol (as described in Current Network Protocols, NIC 7104). 14a

The transmission is to be Network ASCII. The standard receiving mail printer is assumed to have a print line 72 characters wide, and a page of 66 lines. The new line convention will be carriage return (Hex '0D'), (Octal '015') followed by line feed (Hex '0A') (Octal 012') as per the Telnet Protocol, RFC 158, NIC 6768. The standard printer will accept form feed character (Hex'0C') (Octal'014') as meaning move paper to the top of a new page. 14b

It is the sender's responsibility to control the length of the print line and page. If more than 72 characters per line are sent, of if more than 66 lines are sent without a form feed, then the receiving site can handle these situations as appropriate for them. These conventions can be changed by control codes as described below. At the head of the message or document sent there is to be two copies of an initial address string each terminated by a form feed. This address string is to contain the sender's name and address, and the receiver's name and address formatted in some reasonable, easy-to-read form for a clerk to read and distribute. Comments could also be included in the address string. The requirements for two copies are to make one readable from a fan fold paper stack without effort. 14c

Initial Connection 15

Initial Connection will be as per the Official Initial Connection Protocol, Document #2, NIC 7101, to the standard File Transfer socket #3. 15a

File Transfer 16

The mail item (file) to be transferred would be tranferred according to the File Transfer Protocol. 16a

As per the File Transfer Protocol, a file (mail item) can be sent in more than one data transaction as defined in the Data Transfer Protocol. End of file is indicated by the file separator (as defined in Data Transfer Protocol) or by closing the connection. 16b

Order of Transactions 17

The only basic operation required is an Append with Create. 17a

Append with Create Request 17b

(Mailer) User --------------------> Server (Mail Box) 17c

<File - data> 17d

--------------------> 17e

End of File indication 17f

--------------------> 17g

Acknowledge 17h

<-------------------- 17i

The data type default is network ASCII. The standard line printer default is as defined above. Other control transactions can be used. 17j

CONTROL TRANSACTIONS TO BE USED 18

OP CODE 19

Hex	Octal		
			20
09	011	Error or unsuccessful terminate	21
0A	012	Acknowledge or successful terminate	22
05	005	Append with create request (add to existing file or create file if none exists)	23
5A	132	Change printer control settings	24

ERROR CODES 25

All error codes defined in the File Transfer Protocol could be returned. 25a

PRINTER CONTROL CODES 26

Hex	Octal		
			27
D1	321	Meaning: Set line width to 72 characters	28
D2	322	Meaning: Use the full width of your printer	29
D3	323	Meaning: Set page size to 66 lines	30
D4	324	Meaning: Set page size to infinite	31

Other virtual printer control codes can be added in the future. 31a

Other classes of control codes can be added as the need arises. 31b

<WATSON>J8056.NLS;1, 17-NOV-71 14:22 RWW ; Title: Author(s): Richard W. Watson/RWW; Distribution: Steve D. Crocker, Thomas F. Lawrence, John W. McConnell, John F. Heafner, Robert E. Long, Ari O. J. Ollikainen, James E. White, A. Wayne Hathaway, Dan L. Murphy, Patrick W. Foulk, Richard A. Winter, Harold R. Van Zoeren, Alex A. McKenzie, Robert L. Sundberg, James M. Madden, Joel M. Winett, Abhay K. Bhushan, Peggy M. Karp, Thomas N. Pyke, Abe S. Landsberg, B. Michael Wilber, James A. Moorer, Edward A. Feigenbaum, Robert T. Braden, James M. Pepin, Barry D. Wessler, John T. Melvin, John C. LeGates, Art J. Bernstein, C. D. Shepard, Robert F. Hargraves, William S. Duvall, Harvey G. Lehtman/NWG WSD HGL; Sub-Collections: NIC NWG SRI-ARC; RFC# 278; Clerk: RWW;

18516

RFC # 561
NIC # 18516

Abhay Bhushan (AKB) MIT-DMCG
Ken Pogran (KP) MIT-MULTICS
Ray Tomlinson (RST) BBN-TENEX
Jim White (JEW) SRI-ARC
5 September 73

Standardizing Network Mail Headers

1

One of the deficiences of the current FTP mail protocol is that it makes no provision for the explicit specification of such header information as author, title, and date. Many systems send that information, but each in a different format. One fairly serious result of this lack of standardization is that it's next to impossible for a system or user program to intelligently process incoming mail. 2

Although the long-term solution to the problem is probably to add commands for specifying such information to the mail protocol command space (as suggested in RFC 524 -- 17140,), we hereby propose a more quickly implemented solution for the interim. 3

We suggest that the text of network mail, whether transmitted over the FTP telnet connection (via the MAIL command) or over a separate data connection (with the MLFL command), be governed by the syntax below: 4

Example: 4a

```
From: White at SRI-ARC
Date: 24 JUL 1973 1527-PDT
Subject: Multi-Site Journal Meeting Announcement
NIC: 17996

At 10 AM Wednesday 25-JULY there will be a meeting
to discuss a Multi-Site Journal in the context of
the Utility.  Y'all be here.                              4a1
```

Formal Syntax: 4b

```
<mailtext>     ::= <header> <CRLF> <message>                  4b1
<header>       ::= <headeritem> | <headeritem> <header>       4b2
<headeritem>   ::= <item> <CRLF>                              4b3
<item>         ::= <authoritem> | <dateitem> |
                   <subjectitem> | <miscitem>                 4b4
```

```
<authoritem>   ::= FROM: <SP> <user> <SP> AT <SP> <host>           4b5
<dateitem>     ::= DATE: <SP> <date> <SP> <time> - <zone>          4b6
<subjectitem>  ::= SUBJECT: <SP> <line>                            4b7
<miscitem>     ::= <keyword> : <SP> <line>                         4b8
<date>         ::= <vdate> | <tdate>                               4b9
<vdate>        ::= <dayofmonth> <SP> <vmonth> <SP> <vyear>         4b10
<tdate>        ::= <tmonth> / <dayofmonth> / <tyear>               4b11
<dayofmonth>   ::= one or two decimal digits                       4b12
<vmonth>       ::= JAN | FEB | MAR | APR | MAY | JUN |
                   JUL | AUG | SEP | OCT | NOV | DEC               4b13
<tmonth>       ::= one or two decimal digits                       4b14
<vyear>        ::= four decimal digits                             4b15
<tyear>        ::= two decimal digits                              4b16
<zone>         ::= EST | EDT | CST | CDT | MST | MDT |
                   PST | PDT | GMT | GDT                           4b17
<time>         ::= four decimal digits                             4b18
<user>         ::= <word>                                          4b19
<host>         ::= a standard host name                            4b20
<message>      ::= <line> <CRLF> | <line> <CRLF> <message>         4b21
<keyword>      ::= <word>                                          4b22
<line>         ::= a string containing any of the 128 ASCII
                   characters except CR and LF                     4b23
<word>         ::= a string containing any of the 128 ASCII
                   characters except CR, LF, and SP                4b24
<CRLF>         ::= CR LF                                           4b25
<SP>           ::= space                                           4b26
```

Please note the following: 4c

(1) <authoritem>, <dateitem>, and <subjectitem> may each appear at most once in <header>; <miscitem> may occur any number of times. The order of <authoritem>, <dateitem>, and <subjectitem> is insignificant, but they must proceed all occurrences of <miscitem>. 4c1

(2) The case (upper or lower) of keywords -- specifically, 'FROM', 'DATE', 'SUBJECT' ,'AT', <host>, <zone>, <vmonth> and <keyword> -- is insignificant. Although 'FROM', for example, appears in upper-case in the formal syntax above, in the header of an actual message it may appear as 'From' (as in the example), or 'from', or 'FrOm', etc. 4c2

(3) No attempt has been made to legislate the format of <user>, except to exclude spaces from it. 4c3

(4) The time has no internal punctuation. 4c4

(5) No provision is made for multiple authors. 4c5

We recommend that mail-sending subsystems which prefix header information to the text of the user's message be modified appropriately, and that other hosts recommend the above conventions to their users. 5

A Methodology for the Specification of a Message Transport System

Gregor V. Bochmann

University of Montreal

John R. Pickens

SRI International

October 1980

1. Introduction

A growing variety of message systems are presently being designed and implemented. A problem which is reaching major proportions is the interworking between these different systems.[1] In order to make interworking possible, the services provided by the different systems must be comparable. Even if they are not equal, certain adaptations may be devised for transforming one service into another. In any case, it is important to have a precise specification of the service(s) provided by each message system.

As long as the message system consists of one protocol layer with a well defined transmission service for the exchange of protocol data units between the different message system modules, the specification of the service to be provided to the user of the message system may be left relatively vague, although the user may have difficulty adapting to this vagueness. The situation at present is similar to that of the case of the HDLC link layer protocol, which was originally developed without a precise specification of the link layer service. When the modules that used this service had to be defined precisely (for example the network layer over HDLC) then the provided service had to be known precisely.

We note that most existing message systems have essentially one layer of "message protocols" which are responsible for delivering

[1] See [7] for typical functions within such systems

messages to the correct destination. Considering more advanced message system architectures [5, 2] which may include modules for handling name-to-address binding, database servers, media transformation servers, etc., we have a situation where the basic message transport protocol is related to a number of additional system modules that are involved in these different services. It is therefore important that the service provided by the different modules be specified precisely, and at multiple levels of abstraction, so that the operation of other modules can rely on this service.

The object message system of this effort is that of the Draft Internet Message Protocol defined by DARPA [8, 9]. This protocol leaves certain aspects of the provided message service open. This leads to some difficulties, as argued in section 3 of this paper. We then give in section 4 an outline of a possible service specification consistent with the protocol, and in section 5 a formalized specification of this protocol. In section 6 we make a few comments on the Draft Internet Message Protocol on points that came to our attention during our work on its formal specification.

The emphasis of this paper is on the specification technique, and the importance of precise service specifications. We feel that the approaches reported in this paper may be applied to other message system designs as well as the design of other network based distributed systems.

2. Approach

Our specification approach proceeds in the spirit of the evolving specification approaches of the ISO Working Group on Formal Description Techniques [4]. Our specification, therefore, includes a description of the message system "model", followed by the service specification and then by the protocol specification. An interface specification can also be given, though we have not done so, but we recommend it for completeness.

This approach differs from the ISO approach primarily in adopting a hierarchical abstract machine view rather than a layered protocol view. Further study may show the differences to be more a matter of degree than substance.

For specifying the behavior of the modules within such an advanced message system, we have found the model of hierarchical protocol layering to be of limited utility. Rather, a model using abstract machines and step-wise refinement--leading to the specification of machines inside of given machines--has seemed more applicable and flexible. This model has been applied in this paper.

For the specification of a given abstract machine, we consider two parts: (a) the definition of the abstract interfaces over which the machine interacts with other machines. Each abstract interface is defined by the list of different interaction events that may occur at the interface, and possibly some (local) constraints on the order in which they may occur, and (b) constraints (global to the whole machine) determining the possible order and parameter values of the interactions that occur at the different interfaces of the machine.

For the definition of the interaction events and the domain of their parameter values, we found the data type definitional facilities of Pascal and similar languages very useful. For the examples in this paper we have adopted the notation of ADA [1]. For the specification of the global constraints (there are no local constraints in the example considered), we use in this paper some "formalized English". This part of the examples could be specified more formally using either a state-oriented specification language, or a formalism based on execution sequences, possibly including temporal logic [6, 10, 3] or other approaches [11].

3. Model of the Message System

The message system consists of two basic abstract machines, the User Interface machine (UI), and the Message Transport machine (MT). The basic sequence of operations is as follows: A user interacts with his local UI to create a message. The user then instructs the UI to send the message. The UI, in turn, presents the message to the MT for routing and delivery. Eventually the MT presents the message to the destination UI and acknowledges this to the source UI.

The MT (which is internally distributed over different hosts, networks, and gateways) interacts with the UIs (which may reside in different hosts) as shown in the diagram below.

```
#################                                              #################
#               #                                              #               #
#       UI      #                                              #       UI      #
#               #                                              #               #
#################                                              #################
     \        ^                                                    /        ^
      \mail    \ deliver                                          /        /
       \probe   \ ack                                            /        /
        \  -requ.\ probe-response                               /        /
         \cancel  \ cancel-response                            /        /
          v  -requ.\                                          v        /
##################################################################################
#                                                                                #
#                              MT                                                #
#                                                                                #
##################################################################################
```

The specification of the message transport system is given in two steps of successive refinements:

- Global specification of the message transport machine (MT). This leads to the "service specification" of the "message transport layer". (We note that only part of this specification is implied by the document [8, 9].)

- Specification of a message transport module (MTM), which is an integral subcomponent of the message transport machine. This leads directly to the "protocol specification" of the "message transport layer".

The User Interface machine may also be specified in successive refinements, revealing the UIM as the integral subcomponent. We choose to focus on the message transport machine, however. Note that our terminology differs from the reference [8] in that we use UIM/MTM rather than UIP/MPM. Note also that richer decompositions are possible for alternate advanced message systems such as [5, 2].

For each of these specifications we try to give only so much detail as is required for the logical compatibility of the UIMs and the MTMs. Therefore only the logical interaction primitives between the UIMs and the MTMs are specified. Conventions concerning possible grouping of several messages into "bags", encryption and the format and coding of the exchanged messages are independent issues. They are discussed in a separate section. Similarly, the question of UIM-to-MTM and MTM-to-MTM communication (whether through shared memory in a local host or through some communication subsystem) is an independent issue. It is not specified here.

The specification of the MT is an overall description of the way in which the system interacts with UIs at the MT-UI interface. It corresponds to the service specification of [4], if the UI/MT relationship were modeled as a layered protocol hierarchy. This specification is distinct from the MTM specification given in section 3. The latter, which is a refinement of the MT specification, describes both the interaction between the MTM and the local UIMs and the interaction between MTMs for the forwarding of messages.

The MTM-specification does not explicitly include the "end-to-end" properties of the MT specification. It is assumed that these properties can be derived from the MTM specification. Such a derivation constitutes a design verification of the MTM specification (it may also be called a verification of the "message transport protocol").

```
   #################                                               ##################
   #               #                                               #                #
   #     UIM       #                                               #      UIM       #
   #               #                                               #                #
   #################                                               ##################
        ^  \        ^                                                   /         ^   ^
        |   \mail    \ deliver                                         /         /    |
get-    |    \probe    \ ack                                          /         /     |
 new-tn |     \  -requ.\ probe-response                               /        /      |
        |      \cancel  \ cancel-response                            /       /       |
        v       v  -requ.\                                          v       /        v
#########   ##############                                  ##############   #########
# tn-   #   #            #-------> ... ---->#            #   # tn-   #
# distr.#   #    MTM     #  forward         #    MTM     #   # distr.#
#########   #            #<------  ... <----#            #   #########
            ##############          forward  ##############
```

One tn-distributor exists per host[2]

We note that the document [8, 9] concentrates on the MTM specification. However, the following points indicate the importance

[2] The interaction of the tn-distributor with the UIM is different from what is specified in [8, 9]; see section 4)

of a precise and logical MT specification.

- Certain design questions can be answered independently of the message forwarding protocol: How does the UIM indicate the identity of a message to be canceled? When is an acknowledgement of a forwarded message generated? Does the UIM obtain error status and MTM trace information when it receives an acknowledgement?

- Interworking between different message systems seems to imply an agreement on some minimal message service to be provided by all systems. This minimal service may be defined without speaking about message forwarding. It is clear, however, that this service will have an impact on the message forwarding protocol.

- Within the context of a layered system structure including several protocol layers the service specification of a given layer is the basis for the operation of the layer above, which uses the service. It represents also the requirements that must be satisfied by the protocol layer below, which should provide this service.

- The MT specification is required for a verification of the message forwarding protocol.

4. MT Service Specification

The MT specification given below is strongly influenced by the protocol definition of [8, 9], but certain design decisions have been made here in order to clarify the UI-MT interface. We mention in particular:

- Most information fields of forwarded messages are also exchanged with the UIs.
- In particular, the "tid" field (message identifier) is exchanged between the MT and the UI. As indicated in the figure above, the UI obtains unique identifiers from the tn-distributor module of the host and assigns these identifiers to the messages submitted to the MT. This simplifies subsequent interaction between the MT and the UI concerning the same messages, such as delivering acknowledgements or indicating probe or cancel responses. In order to "understand" these responses the UI needs to know the identifiers associated with the original requests.

We have structured the specification of the MT into the following three parts:

- specification of data types (corresponding to the "objects" of [8, 9]) which characterize the information exchanged between the message transport system (MT) and its users (UIs),
- specification of the service primitives that may be invoked for the exchange of information between the MT and one of its users, and
- global constraints on the execution of service primitives with different users (end-to-end properties).

4.1 Data types

The following is a specification, in ADA, of the structure of messages that are passed between the UI and MT. This structure is also utilized internal to the MT.

```
TYPE message IS
    RECORD
        tid : transaction-id ;
        destination : mailbox ;
        trace : MT-list ;
        command : (deliver, acknowledge, probe, response,
                   cancel, canceled) ;

        CASE command OF
            WHEN deliver =>
                options : delivery-options ;
                document : RECORD
                               header  : header-type ;
                               body : LIST OF body-portion ;
                           END RECORD ;
            WHEN acknowledge =>
                acknowledged-tid : transaction-id ;
                trail : MT-list ;
                answer : boolean ;
                CASE answer OF
                    WHEN  true =>
                        delivered-address : mailbox ;
                        delivery-mode : (regular, gendel) ;
                    WHEN false =>
                        reason : why-not-delivered ;
                END CASE ;
                how-delivered : delivery-options ;
                status : error-list ;
            WHEN probe => NULL ;
            WHEN response =>
                request-tid : transaction-id ;
                trail : MT-list ;
                answer : boolean ;
                CASE answer OF
                    WHEN true =>
                        address : mailbox ;
                    WHEN false =>
                        reason : why-not-delivered ;
                END CASE ;
                status : error-list ;
            WHEN cancel =>
                to-be-canceled :transaction-id ;
            WHEN canceled =>
                canceled-tid : transaction-id ;
                trail : MT-list ;
                answer : boolean ;
                status : error-list ;
        END CASE ;
    END RECORD ;
```

```
TYPE MT-list IS  ... ;
TYPE delivery-options IS SET OF (regular, forward, gendel);
TYPE why-not-delivered IS (no-such-user, no-such-host,
          no-such-network, address-ambiguous, mailbox-does-not-exist,
          mailbox-full,  ... ) ;
TYPE error-list IS
    RECORD
        error-class : (success, partial-results-returned,
            resource-unavailable, user-error, MT-recoverable-error,
            MT-fatal-error, user-abort-requested) ;
        error-string : text ;
    END RECORD ;
TYPE text IS  ... ;
```

4.2 Service Primitives Executed at the UI-MT Interface

```
[snowpake]
    post  (m : message (command => deliver) )
    deliver (m : message (command => deliver) )
    ack (a : message (command => acknowledge) )
    probe-request (p : message (command => probe) )
    probe-response (r : message (command => response) )
    cancel-request (c : message (command => cancel) )
    cancel-response (c : message (command => canceled) )
```

[3]

We assume that these primitives are executed jointly by the two modules involved in the interface. The primitives post, probe-request, and cancel-request are initiated by the UI, the other primitives are initiated by the MT. The value of the parameter is determined by the initiating module. For the former, the trace field has the value empty; for the latter, the destination field has as its value the mailbox of the UI involved in the execution of the

[3] This line reads: There is a "post" primitive which requires a parameter of type "message" (defined in section 2.1) for which the "command" field has the value "deliver".

primitive.

4.3 Global Constraints

The following rules specify constraints on the order of execution and on the possible parameter values of the primitives executed at different places within the message system ("end-to-end" properties). The first rule states that every message sent will be eventually delivered, while the second states that each delivery will be acknowledged. Similar rules must also be given for the probe and cancel primitives. As they stand, these rules are incomplete, since the handling of exceptional cases is not specified.

1. After the execution of the primitive post(m) by some UI and the MT, where m.destination = some value x, there will eventually be an execution of a primitive deliver(m') by the UI with the address x and the MT, where m'=m except for the trace field; the first element of m'.trace = the ihn (internet host number) of the host where the post(m) was executed. This is the only case in which a deliver primitive will be executed.

2. After the execution of deliver(m') primitive by some UI and the MT, where the first element of m'.trace is the ihn y, there will eventually be an execution of an ack(a) with the host ihn = y, where a.acknowledged-tid = m'.tid, a.trail = m'.trace, and a.answer = true. This execution will be related to the mailbox of the UI which had previously executed the post(m) primitive with m.tid = a.acknowlecged-tid. This is the only case in which an ack primitive will be executed.

4.4 Addressing

```
TYPE mailbox IS
     LIST OF RECORD
          property : (ia, net, host, user, city, ... , phone) ;
          CASE property OF
               WHEN ia       => internet-address : integer ;
               WHEN net      => network-name : text ;
               WHEN host     => host-name : text ;
                ...
          END CASE ; END RECORD ;
```

5. Specification of the MTM-level Protocol

We use in the following the same specification approach and subsections as in the case of the MT specification above. The interactions of an MTM module are indicated in the second diagram above.

5.1 MTM Interaction Primitives

The following primitives are executed by the MTM:

- Primitives executed in relation with UIs (see section 4.2)
- forward (m : message) This forward primitive is used by the MTMS to forward messages towards the destination MTM. The "message" type is defined in section 4.1.

5.2 Constraints

The following specifies constraints on the execution order and on the possible parameter values of the interaction primitives executed by a given MTM:

1. For each incoming message m with command = deliver (coming in through a post primitive from a local UI, or through a forward primitive from a remote MTM) do the following:

```
        IF is-local(m.destination)
        THEN execute deliver(m') with the corresponding UI where
                   m'=m, except that m'.trace = m.trace appended with the
                         host ihn (internet host number) of the MTM ;
             put an acknowledgement message a into the handling queue
             (for the same MTM)  where
                         a.acknowledged-tid = m.tid,
                         a.trail = m.trace, and
                         a.answer = true ;
        ELSE find-route(m.destination, next-MTM) ;
             execute a forward(m') primitive with the next-MTM
                   where m' = m , except that
                         m'.trace = m.trace appended with the ihn of the MTM
```

For each incoming acknowledgement ....

...

...

Note: The above specification is incomplete, because it does not consider the cases where the message cannot be delivered.

5.3 MTM-to-MTM and UIM-to-MTM Communication

Several additional layers of protocols must be specified for the MTM-to-MTM and UIM-to-MTM communication. This includes a specification of the underlying transport service, as well as protocols for message bagging, data compression, and encryption. Also the format and coding of the message type structure (see section 4.1) must be specified.

For the MTM-to-MTM communication these protocols must be standardized for all systems involved, while the protocols to be used between a given UIM and its MTM may be agreed upon by the two parties (often this will be conventions within one single computer operating system). The chosen protocols, however, must meet the requirements of the service specification.

5.4 Message Bagging

Instead of initiating an interaction with a peer MTM for each message to be forwarded, several messages may be combined by an MTM into a bag, and a whole bag of messages may be forwarded between two MTMS (at an appropriate time) in a single interaction.

The same bagging strategy may also be used for the exchange of messages between a UIM and an MTM, although this is a question of implementation of the UIM-MTM interface, which has no impact on the UIM-to-UIM compatibility (i.e. a pair of UIM's are compatible even if one does not support bagging).

The bagging strategy described in document [8, 9] may be characterized by the following rule: Whenever a given data structure, say "some-data", is possibly to be shared between several messages the field "some-data" is replaced by a data structure of the form

```
shared-data : RECORD
                        sharing : boolean ;
                        CASE sharing OF
                            WHEN true =>
                                tid-to-share-from : transaction-id;
                            WHEN false =>
                                some-data : ... ;
                    END RECORD ;
```

The complete specification which includes bagging may be found in [12].

6. Comments on the Internet Message Protocol

The following issues have arisen from our efforts to specify and evaluate the message transport system. We feel that the more formal and concise specification has helped to facilitate the detection of problem areas.

- The naming - addressing scheme of the protocol is host and network oriented, which corresponds to the addressing scheme of an interprocess transport protocol. Alternate naming schemes, such as organizational hierarchy, can be expected to be processed by both the UI and the MT.

- There is in general no information on the source of a message available to the receiving UIM and its MTM, except that the message header may include some source identification which, however, has only an informal meaning. This has the following consequences:

  1. The final MTM receiving an acknowledgement has some problem delivering it to the right UIM (unless it keeps a list relating the transaction numbers of the sent messages to the identity of the sending UIM's).

  2. A UIM module of user x may masquerade as user y (as seen by the receiver who reads the message header), unless the posting MTM checks the information inside the message header (which is usually not looked at by an MTM).

- The meaning of an acknowledgement is not clear as far as its "end-to-end" significance is concerned. It is not clear who generates the acknowledgement, the receiving UIM, the message server module of the UIM (which may be on a local network), or the last internet MTM through which the message travels. We assume the latter. In this case the message may not yet have reached the destination UIM when the acknowledgement is generated.

- It seems that for certain applications more flexible sharing arrangements within bags of messages would be useful, and should be foreseen. An example would be a structured message body consisting of several components where not all of these components are to be shared.

- The UIM needs to know the "tid" identifiers associated with the requests it passes to the MT. This is necessary for interpreting acknowledgements, and probe and cancel responses returned by the MT. It is also necessary if the UIM submits bagged messages.

- The dialogue between MTMs is simple, and probably should be expanded. Two examples of needed inter-MTM dialogues which go beyond the basic "forward" primitive include change-of-address and flow-control. Change-of-address allows a message to be rerouted before transmission of the entire body, an important efficiency consideration. Flow-control may be needed in order for an individual MTM to restrict the message traffic flowing in or out of a specific path.

7. Conclusions

In this paper we have discussed a methodology for precisely specifying a message system. We have argued that such a specification must include not only the protocol(s) used by the different components of the system, but also the specification of the overall service provided to the users of the system.

In the context of advanced message systems with host-independent naming , specialized media etc. , the specification of each of the different system components must be given. The "model" of the message system which we use is not necessarily structured into a number of protocol layers; but rather a more flexible structuring principle is used where a given system modules may be subdivided into a number of submodules. The message transport system, for example, is considered as an abstract machine, the specification of which is the specification of the message service provided to its users. Subsequent subdivisions leading to more and more detail are then considered. One such division leads to the consideration of the MPM, the specification of which defines the message transport protocol.

The message service specification is given in terms of service primitives, which are also referred to by the protocol specification. Other advanced functions could be specified separately, defining additional modules.

Finally, we have found that our efforts in making the service and protocol specifications more precise have led us to a better understanding of that system, and to the discovery (or observation)

REFERENCES

[1]

Ichbiah, J. D., et al.
Preliminary Ada Reference Manual and Rational for the Design of the Ada Programming Language.
Sigplan Notices 14(6), June 1979.

[2]

Lebling, P. and S. Galley.
Advanced Message System Design Note.
Technical Report AMS.01.01, Programming Technology Group, Laboratory for Computer Science, Massachusetts Institute of Technology, April 21, 1980.

[3]

Hailpern, B., S. Owicki.
Verifying Network Protocols Using Temporal Logic.
Technical Report Technical Report 192, Computer Systems Laboratory, Stanford University, June 1980.

[4]

Guidelines for Specifications of Services, Protocols and Interfaces for OSI.
ISO/TC97/SC16/WG 1.

[5]

Levin, R. and M. Schroeder.
Transport of Electronic Messages Through a Network.
Teleinformatics 79 :29-33, 1979.

[6]

Manna, Z., A. Pnueli.
The Modal Logic of Programs.
Technical Report STAN-CS-79-751, Computer Science Dept., Stanford University, September 1979.

[7]

Pickens, J.R.
Functional Distribution of Computer Based Messaging Systems.
In Proceedings of Sixth Data Communications Symposium, . ACM/IEEE, November 1979.

[8]

Postel, J. B.
Internet Message Protocol.
Technical Report IEN 85, USC Information Sciences Institute, Marina Del Rey, CA, March 1979.

[9]

Postel, J. B.
An Internetwork Message Structure.
In _Proceedings of Sixth Data Communications Symposium_, .
ACM/IEEE, November 1979.

[10]

Schwartz, R. L. and P. M. Melliar-Smith.
Temporal Logic Specification of Distributed Systems.
August 1980.
Computer Sciences Lab, SRI International, submitted for publication.

[11]

Bochmann, G. V. and C. A. Sunshine.
Formal Methods in Communication Protocol Design.
IEEE Transactions on Communications 28(4):624-631, April 1980.

[12]

Bochmann, G.V. and J.R. Pickens.
A Methodology for the Specification of a Message Transport System.
Technical Report SRI/TSC/80-1, Telecommunications Sciences Center, SRI International, July 1980.

Table of Contents

TOWARDS VIDEOTEX STANDARDS

by

Gregor V. Bochmann and Jan Gecsei

Publication # 350

Département d'informatique
et de recherche opérationnelle
Université de Montréal

novembre 1979

ABSTRACT

Videotex is a generic name for a class of information services based on a modified home TV set with a keypad for user feedback. While present system designs center around a page-oriented database for information retrieval of interest to the general public, future videotex systems will probably evolve towards a generalized information utility with innumerable applications. In order to enable orderly growth of videotex technology, a framework of standards and conventions is needed. This paper attempts to identify the functions in videotex systems, for which standard communication protocols will be needed. These functions are discussed on the background of the ISO reference model for open systems interworking.

1 - INTRODUCTION

Videotex is the generic name for a class of information services based on a modified home TV set with a keypad for user feedback. This terminal is connected through various transmission media to a computer providing the necessary processing and database services. The most often used transmission channels in todays systems are telephone lines and the TV transmission.

Two broad categories of systems can be recognized:

(a) One-way systems, where interaction exists only between the user (keypad) and the terminal, and

(b) two-way systems with full interaction between the user and service computer. Except for the limitation of the usual keypad (which can be easily overcome), there is little difference between the potential of such interactive videotex systems and any general purpose computer system. The actual difference is that videotex is intended to cater to the general public; therefore its initial cost and complexity should be acceptable to the non-expert users. As a result, the initial application chosen in practically all systems is retrieval of information of general interest from public, page oriented databases. However, it should be kept in mind that in the future videotex will probably evolve towards a generalized information utility with innumerable and sometimes unforseeable applications.

Functional and technical descriptions of many present systems are available in the literature [1], [2].

It can be seen that videotex technology involves a large number of interworking components such as terminals, TV sets, telephone networks, national and international data networks, computers and databases. These components are designed, manufactured and operated by many organizations. In order to enable orderly growth of videotex technology, a framework of standards and conventions must be provided. Some recent proposals in this direction are e.g. for a terminal-independent scheme of information coding

[1], and for layered terminal capabilities [3].

The present paper is an attempt to identify the functions of a videotex system, for which standard protocols will be eventually desirable. Some of these protocols should be standardized soon, while other functions are not yet well understood at the present. We take the ISO model for Open Systems [4] as a reference for our discussion.

2 - THE ISO REFERENCE MODEL FOR OPEN SYSTEMS

The subcommittee SC 16 of ISO TC97 has defined a standard reference model [4] for the architecture of systems that are "open" for interworking with other systems in a distributed environment. The model distinguishes several layers of communication protocols. The functions provided by the different layers may be characterized as follows:
(1) Physical layer: transmission of unstructured bit sequences.
(2) Link layer: control of logical links between systems components, data transmission with error recovery.
(3) Network layer: routing through networks, circuit switching, virtual packet-switched circuits.
(4) Transport layer: logical connections between processes in different computers, uniform data transport service independent of the underlying physical transmission medium.
(5) Session layer: control of sessions over logical connections.
(6) Presentation layer: representation and coding of data structures, such as characters, display formats, etc.
(7) Application layer: "applications" using the services provided by lower layers, distributed system management such as collecting statistics on traffic, reporting of failures, etc.

While the lower layers of the architecture are relatively well understood and exemplified by existing communication protocol standards, the higher levels are presently under study for the development of international protocol standards. Among these standards being developed, the Virtual Terminal protocol is closely related to the videotex terminal functions discussed below.

3 - PROTOCOL FUNCTIONS FOR VIDEOTEX APPLICATIONS

We give in this section a preliminary list of functions that must be handled by the database system through interaction with the terminals, other database systems, and possibly administrative entities. These interactions are governed by appropriate communication protocols. We give an explication of each function, and indicate in which layer of the ISO reference model it would probably reside. If appropriate we also mention existing protocols that may be used for these functions in the videotex framework. It is clearly very important, in adopting standards for the present simple videotex applications, to avoid the dangers of early standardization [3], and ensure upward compatibility with future more sophisticated applications.

Most of the protocol functions mentioned below involve the database system and the user's terminal. It is important to note that, depending on the intelligence of the terminal, many of these protocols may operate between the database on one side and the terminal or the user on the other side, or between the terminal and user.

As an example, consider browsing through classified car ads in order to find a used Datsun car. With a simple terminal, the user will sequentially read through the ads, giving signals to proceed through the data. A terminal with plug-in intelligence might perform a "Datsun" keyword search for the user, searching automatically through the car ads data segment and displaying only the relevant ads. Alternatively, the same intelligence may be down-line loaded from the database to the terminal, if the latter has telesoftware capability. Apart from the telesoftware protocol, the database system executes a simple browsing protocol interacting with the user or terminal. In the case of intelligent terminals, the user executes a keyword search protocol (defined as a sequence of user commands and system responses) which will be specific to the plug-in intelligence or telesoftware. Such a keyword search protocol may also be executed directly between a user and database if the latter supports such a facility. Only a simple terminal with character feedback would be needed.

3.1 - Data transmission and session control

The functions described below must be supported by an appropriate data transport service and session control between the database and other components, especially the terminals. These functions are provided by the session layer and the layers below it in the ISO model. We do not discuss these functions here.

3.2 - Protocols for terminal operation

The following functions are part of what might be called a videotex virtual terminal, and belong mostly into the presentation layer of the ISO model.

3.2.1 - Display control

There are two aspects of data display that must be controlled: (a) the structure of displayed data, and (b) the temporal order in which parts of an information segment are displayed (in simple systems this order is page by page).

For transmission, the data to be displayed must be coded. Compactness and independence from terminal display characteristics are important objectives for the choice of appropriate coding schemes. Depending on the type of information, different schemes may be used, such as the following:

For textual information:
- standard ASCII character set
- ISO code extentions (e.g. for alpha-mosaic drawings)
- display format controls (e.g. for tables, annotation of diagrams)

For graphical information with gray scale and colors:
- Telidon PDI [1]

- ACM graphics standard [6].
For photographic information:
- Telidon PDI
- facsimile
For voice information:
- PCM
- compressed voice
For video imagery (moving images):
- analog storage and/or transmission may be used, which lead to a hybrid system (analog for video and digital for data and control).

All these coding schemes would belong to the presentation layer of the ISO model.

Another aspect is the temporal order in which different parts of an information segment may be displayed. We distinguish the following cases:
(a) Simultaneous display of the whole segment as a whole page.
(b) Sequential display if the segment contains several pages. Roll mode (suitable for general text information) and page mode (for information organized as a sequence of display pages) are available.
(c) Incremental display, where elements are added to an evolving picture.
(d) Interactive selection of complex display sequences, such as in the case of interactive manuals, courses, etc.

3.2.2 - Telesoftware [7]

This function provides the possibility of loading programs and data into the terminal and initiating the execution.

3.2.3 - User feedback

Retrieval applications usually involve at least numerical and special function information feedback from a simple keypad. More sophisticated applications may involve text and possibly some graphic interaction, such as joystick or mouse.

3.2.4 - Option negotiation

Option negotiation is the process of selecting the facilities and protocols that are to be used during a subsequent communication session. The negotiated facilities and protocols will depend on the application to be performed and on the capabilities of the involved terminal and database. This may include different display types, telesoftware options and feedback options to be used, particular selection functions (see below) to be provided by the database, and the physical limits of the terminal such as screen size, graphical capabilities, memory size for telesoftware, etc. A possible protocol scheme for negotiation is described in [8].

3.3 - Information retrieval

The database structure determines the procedures by which a user may retrieve information on the terminal (part of the ISO presentation layer).

Retrieval involves user feedback and different selection functions available in the database. Present videotex databases usually have a tree-like structure, where each node of the tree contains an information segment which itself may consist of a number of displayable pages. The so far implemented selection functions include the following:
(a) Direct selection of a data segment or page by a user-given identifier.
(b) Multiple choice procedures, where the user responds to a displayed "menu" by choosing from a fixed set of alternatives.

These functions involve only segment and page identifiers, but not the information content of pages. More powerful selection procedures are desirable for future applications, such as:
(c) Keyword searches (a possible protocol is described in [9]).
(d) General queries in formatted files and databases (a screen oriented query language is described in [10]; different approaches to the use of natural language are discussed in [11]).

These selection procedures involve the content of data segments, and usually require more processing in the database computer. A more detailed discussion of possible database structures and selection procedures for videotex may be found in [12].

3.4 - Cooperation between databases

Future videotex systems will probably include several different databases cooperating in a varying degree. In one extreme, they may be separate independent databases; in the other extreme, they logically represent a single database. We assume here that several databases cooperate to present to the user a data structure which makes largely abstraction from the distribution of the data into different databases. The latter may contain each specialized information, as well as multiple copies of the same data.

Figure 1 shows as an example a global directory, as seen by the user, involving data in different databases. The upper part of the directory information may be replicated in several local videotex service computers, while the provincial and corporate information may only exist in one copy. The example also shows how different views of the data may be obtained depending on user's viewpoints. Through the "public DB" access point, the only accessible data about "NT" is the "NT overview" and its subtree, while other information, such as "personnel" is visible through the "NT corporate DB" access point.

Cooperation, as in this example, clearly requires certain similarities and communication standards between the databases involved. At the present it is not clear, what kind and how much cooperation would be useful and feasible for videotex applications. Nevertheless, we may identify the following functions, which may be allocated to the ISO application layer.

3.4.1 - Cooperation for retrieval

Database cooperation may include the following functions for making information retrieval more efficient and/or simpler for the user:
(a) Forwarding: a segment or page requested by the user may not exist in the local service computer; it may be forwarded on demand from another database.

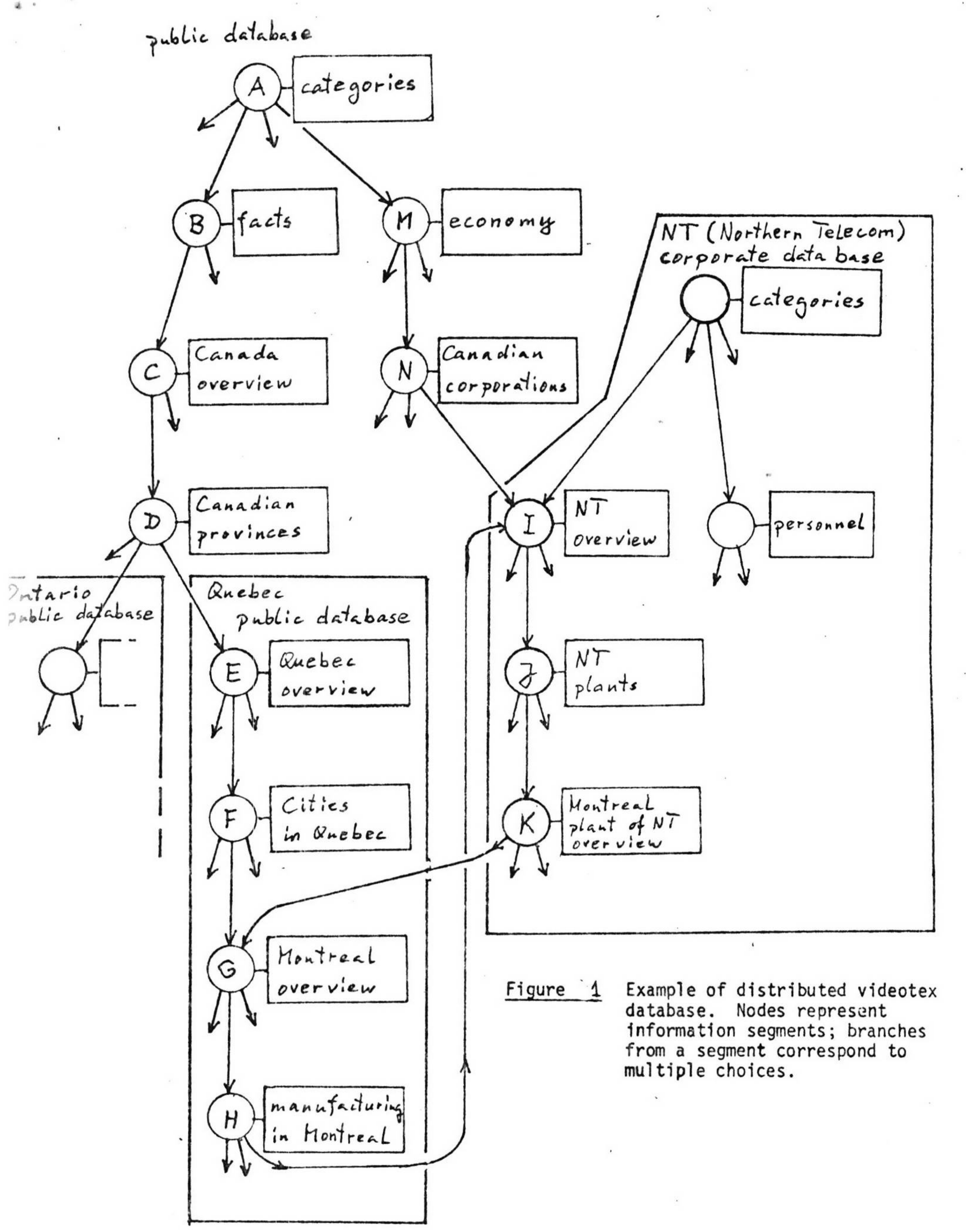


Figure 1 Example of distributed videotex database. Nodes represent information segments; branches from a segment correspond to multiple choices.

(b) Local copies: It is advantageous for efficiency to keep local copies of frequently accessed pages.
(c) Superdirectory: A directory of all available databases and other services may help to transfer the user directly to the chosen service computer.
(d) Global directory: As shown in the example of fig. 1, cooperation of different databases may lead to a single global directory, as seen by the user.
(e) Multiple views: As in the example, different access points may give the user different (partial) views of the stored data.

3.4.2 - Updates

Database updates involve the following two phases: (a) the creation or update of a segment, and (b) its incorporation or deletion into (from) the database, with a corresponding simultaneous update of the directory. Phase (a) will usually be executed by an editor (terminal) which may obtain an existing segment through a forwarding protocol. Phase (b) presents the problem of keeping the database directory consistent with the available information segments. Directory structures such as in fig. 1 are certainly not easy to keep consistent because of the many possible relations between segments. These consistency problems become even larger if global directories for several databases are involved.

An important function of the update protocol in cooperating databases is thus to keep the directories consistent with one another. The protocol will depend on the database structures used and their cooperation, and on the degree of consistency required.

3.5 - Distributed system management

Here we consider protocols for the following functions, which might be allocated to the ISO application layer:
(a) User identification: This function is needed to verify access rights (except for public databases), and for charging.
(b) Verification of access rights: (not needed for public databases).
(c) Charging: We can identify the following groups, companies or administrations involved in the operation of videotex systems: the user, the terminal provider, the transmission provider, the database administration and the information provider.

Usually the user, and sometimes the information provider (e.g. in advertizing) will pay for the services provided for him by others. Simple protocols must be provided to automatically execute the appropriate financial transactions. Third party billing schemes are essential, see e.g. [13].
(d) Data distribution: This involves operating, monitoring and controlling updates in distributed databases and the management of global directories.

3.6 - User support and help facilities

A friendly user interface is important for the acceptance by casual users of videotex services. The procedures for getting information about how to use the services offered, should, as far as possible, be the same for all services [9]. Such procedures should include

(a) a general HELP command,
(b) getting a list of available commands and their meaning,
(c) getting an explanation of each command, its function, and eventually examples of its use,
(d) interactive choice of parameter values with prompting the name of the next parameter, etc.

As an example, many of the above functions are implemented in the UNIX operating system [14].

REFERENCES

[1]...H.G. Bown et al.,"Telidon: A New Approach to Videotex System Design", IEEE Trans. on Consumer Electronics, Vol. CE-25, No 3 July 1979, p. 256.

[2]...R. D. Bright, "Prestel, the World's First Viewdata Service", ibid., p.251.

[3]...J. M. Costa and A. H. Marsh, "Videotex System Planning", Proc. National Electronics Conference, Chicago, Oct. 1979, p.337.

[4]...ISO, TC97/SC16 N 117, "Referance Model for Open Systems Architecture", Nov. 1978.

[5]...J.C. Smirle and H. Bown, "New Systems Concepts and their Implications for the User", Proc. First CCITT Symp. on New Telecommunications Services, Geneva, May 14-16 1979, p.128.

[6]...J.C. Michener and A. Van Dam, "A Functonal Overview of the Core System with Glossary", ACM Computing Surveys 10, 4, Dec. 1978, p.381 (and other articles in the same issue).

[7]...J. Hedger, "Telesoftware", Wireless World, Nov. 1978.

[8]...IFIP WG 6.1, "Proposal for a Standard Virtual Terminal Protocol", INWG Prot. Note 91, Febr. 1978.

[9]...A.E. Negus, "Euronet Guideline: Standard Commands for Retrieval Systems" (report for the Commission of European Communities DGXIII), Dec. 1977, INSPEC, Savoy Place, London, WC2R OBL, England.

[10]..IBM, "Query by Example", Report SH 20-2077.

[11]..D.L. Waltz, "Natural Language Interfaces", ACM Sigart Newsletter 61, Febr. 1977, p.16.

[12]..J Gecsei and G.v. Bochmann, "Database Structures in Videotex Applications", submitted to IFIP Congress 1980, Tokyo.

[13]..I. de Sola Pool, "International Aspects of Computer Communications", Telecommunications Policy 1,1, Dec. 1976, p. 31.

[14]..D.M. Ritchie and K. Thompson, "The UNIX Timesharing System", Comm. ACM, 17, 7, July 1977, p. 365.

BOLGER.

01/80

executive editor's notebook

One of the more interesting projects the editorial staff of INFOSYSTEMS has ever undertaken was to prepare the articles for "The Computer In Our Lives" series in this issue. Here was an opportunity to stand back and examine the impact the computer is having on us as people and as consumers. As editors of a user magazine in this industry, we're constantly writing about computers. But in a matter-of-fact, business-like way. We may have become inured to what depths this marvelous device, the computer, is reaching.

So the assignments were made last fall for each of the eight editors assigned to this project to fan out across the US and to take a look from a consumer's viewpoint what the impact has been. Needless to say, the outcome was probably predictable—the computer's impact has been enormous. The 11 pages given to the series can only scratch the surface of the computer's involvement in our lives. At the risk of being irreverent, it affects us from "dust to dust." Just one facet of our lives alone—the government—assures this. Vic Block, our Washington editor, points out that the US government is the world's largest user of computers. The government's computers count us, guide us in our travels, protect us and pay many of us, to name a few examples.

It would be difficult to pin-point the one area where the computer has the greatest effect. Our Southwest Editor Steve Stibbens might say in the growing use of transferring funds electronically. Neil Kelley might say anybody who ever made a transaction has his or her credit record on file. Wayne Rhodes would explain the growing use of computers in medicine. However, Carla Schanstra would quickly point to the growing use of the computer in the office.

No one would disagree, though, that the computer's impact is great and will increase in quantum jumps in the months and years to come. It was a fun series. If you like it, we'll do it again someday. —John M. Lusa

reader feedback

Readers are invited to express themselves on any subject related to the infosystems industry, or to the use of infosystems. Address: Letters Editor, INFOSYSTEMS, Hitchcock Building, Wheaton, IL 60187

Postmaster General replies

I have read the editorial you sent to me that appeared in the September issue of INFOSYSTEMS on President Carter's decision to allow a role for the Postal Service in the electronic mail field.

It brought to mind the "Lawyer's Rule," which says: "When the law is against you, argue the facts. When the facts are against you, argue the law. When both are against you, call the other guy names."

While you have a right to your opinion, your use of such terms as "past record of ineptitude" and "multibillion dollar boondoggle" is not only inaccurate and unfair, but does nothing to advance a rational dialogue on this subject.

Your contention that the Postal Service is not equipped to move into the electronic field ignores the fact that we successfully have offered "MAILGRAM", our first electronic service, since 1970. A joint venture with Western Union, "MAILGRAM" volume and revenue have grown at an annual rate of 20 percent over the past five years. The fact that it is a joint USPS-private industry venture underscores President Carter's concern (and our long-held position) that electronic mail should be offered jointly by the Postal Service *and* private firms.

I made this point abundantly clear in my statement at the time of the President's policy statement on Postal Service involvement in electronic mail. In addition to saying "the President has given us the opportunity to provide an improved level of mail service at lower costs to the public," I went on to observe: "And in stating that the Postal Service should use the

. . . continued on page 10

vol 27, no. 1

. . . continued from page 6

transmission services of the private carriers, he has insured that the interests of the private sector will not be infringed."

I have repeated this position in many forums and say again that our purpose and declared role in electronic mail has been and will continue to be to link the Postal Service's "hard copy" delivery capabilities with the telecommunication capabilities of private firms.

Our limited monopoly over First-Class and other letter mail would not extend to the electronic transmission of messages, only to the delivery of the "hard copy" messages at the receiving end.

Your editorial contains the contradictory criticisms that the Postal Service should be excluded from the electronic area because it is inefficient, but in the next breath you assert that our new services like Express Mail are actually too competitive for your taste. You attribute their success to being subsidized.

Yet, the facts are that each class of mail must pay its own attributable costs and a portion of the Postal Service's fixed costs. There is no cross subsidization and the use of these services by the public and business firms is based on their good service performance, competitive price and convenience.

While the Postal Service does receive Congressional funds, the level of that support has declined sharply from 24 percent in 1971 to nine percent in 1979. A large portion of the subsidy goes to support lower rates for certain classes of mail as decided by Congress and this is not a subsidy to the Postal Service. The rest is a public service subsidy that helps support the costs of all of our activities. Incidentally, if the entire subsidy was eliminated it could only mean an increase of one cent on the letter rate.

Your criticisms of the Postal Service's performance hark back to the postal system of the past that has been transformed into a modern, responsible government service since postal reorganization took effect in 1971.

Through mechanization and modern management, we have reduced our payroll through attrition by 80,000 employees in 1978 over 1971, while volume increased during that period by almost 12 billion pieces.

Our basic bread-and-butter First-Class service has not been given a back seat to new services like Express Mail and "MAILGRAM," as you charge. We regularly deliver overnight 95 percent of stamped, First-Class Mail that qualifies for next day delivery.

All these improvements are most clearly reflected in the annual net income the Postal Service will have for the fiscal year ending September 30, the first such annual net income in 34 years. This performance should allow us to go 2 1/2 years between rate increases, a record few firms in the private sector can match in these inflationary times.

The Postal Service involvement in electronic mail is well within our historic tradition of marching to the beat of advancing times. We have progressed from stagecoach and steamboat to railroads, trucks and airplanes to move the mail. We see the use of electronic mail in that tradition, with the promise of benefits to the public, the private sector and the Postal Service. It will sharply reduce the handling costs and help stabilize

. . . continued on page 12