SPECIAL ISSUE ON EIN AND OTHER EUROPEAN COMPUTER NETWORKS
ALTA FREQUENZA

CONTENTS

Foreword .................................................................................................................. 468 - 244 E
A decade of development in computer communications - D. W. Davies .................................................................................................................. 469 - 255 E
EIN - An example of cooperative research in Europe - D. L. A. Barber .................................................................................................................. 472 - 258 E
Gateway on higher level protocols - P. Schlebar - A. Daurati .................................................................................................................. 498 - 321 E
The EURNET network: origins, reasons, and possible future applications - N. Bolognini - A. Maino .................................................................................................................. 504 - 330 E
Network management and control in EURONET - A. M. Reggiani .................................................................................................................. 511 - 332 E
SARA: a network between non-homogeneous batch computers - C. Di Filippo - P. Masegno - A. Matteuzzi - G. Mettoni - M. Zazzoli .................................................................................................................. 531 - 357 E

From R & D Laboratories:


Some measurements on the EIN computer network performed at CREI by means of the subnetwork control module - A. Faro - G. Scallo - F. Valova .................................................................................................................. 542 - 368 E

CONTRIBUTORS

552 - 279 E

C O N T R I B U T O R S

It shows in enlarged scale the main constructional features of our miniature tendril. On assembling, sliding of the metal part within the insulation creates a pre-assembling which uses particularly well modern assembly techniques.

Manufacturer: FERRE - Genova (Italy)
**CONTENTS**

- Foreword
  - A decade of development in computer communications
  - EXN: An example of cooperative research in Europe
  - RPRESSNET: Status and trends
  - SABRE: A network between non-homogeneous batch computers
  - GATEWAY on higher level protocols
  - Host-terminal connection techniques in networks
  - The EURUNET network: origins, reasons, and possible future applications

**Contents**

- Network management and control in EURO-NEAT (A. M. Rehnhem)
- A formal description of the DTE data unit in the X.25 Recommendation (S. Caforio - A. Formenti)
- GATEWAY on higher level protocols (P. Schicketanz - A. Denti)
- The EURUNET network: origins, reasons, and possible future applications (M. Mangani - A. Milito)

**network management and control in EURO-NEAT (A. M. Rehnhem)**

- A formal description of the DTE data unit in the X.25 Recommendation (S. Caforio - A. Formenti)
- GATEWAY on higher level protocols (P. Schicketanz - A. Denti)
- The EURUNET network: origins, reasons, and possible future applications (M. Mangani - A. Milito)
A decade of development in computer communications

D.W. Davies
NATIONAL PHYSICAL LABORATORY, TEDDINGTON MIDDLESEX

Abstract. A brief look at the early development of packet-switched computer communications, the paper considers some critical factors in the development of a satisfactory network - the human interface, reliability and availability of service, and the possibility of change in both technology and in user requirements. Packet switched networks illustrate how the movement of intelligence towards the network periphery was felt in all these respects. The paper concludes with a glance at the future of public networks.

1. INTRODUCTION

The European Information Network project is the subject of several papers in this issue of Alta Frequency. This project has been a significant part of the first decade of so of research and development network which has brought the packet switched computer network from the laboratory to every day use.

The way in which EIN has evolved illustrates the principle that the communication system should be both simple and the network's intelligence should be at its periphery. In this paper the critical factors in network development are related to this trend towards peripheral intelligence.

2. EARLY DEVELOPMENT OF COMPUTER COMMUNICATIONS

In his introductory paper to the EIN proceedings Paul Baran and his team at Rand Corporation in 1961 described a distributed network with a short, standard message block and adaptive routing. Their main concern was the need for 'networks' which were to be more efficient than the telephone and almost instantaneous recovery of communication. They did not design their system specifically for computer communications, but they needed it to transmit both data and digital voice. Even at this early stage, end-to-end, digitised voice communication was described in some detail. The paper by Green and Frank in the same issue showed that packet voice is emerging again as an economic possibility.

The special affinity of packet communication with computers and terminals was brought out by two subsequent projects. In my own laboratory, from 1960 onwards, we built a packet switched network for local communications. The network was a mesh of four new nodes. We also extended it as an extensive series of simulations of routing and flow control by computer simulation. In a paper of June 1960, which recorded a lecture I gave in March 1960, the word 'packet' first appeared - a word which I chose carefully to be translatable but which caused trouble in Russia. The APA network project was planned in 1961 and began operation in 1962, giving the first experience of a large-scale network based on packet switching and introducing a whole generation of researchers to these techniques.

Another important European development began in 1960, the ITIN high level network, combining packet and message switching. Shortly afterwards there was the TELELINES project, the first wide area network to exploit the form of packet switching which became known as a 'datagram' network and to build it from a reliable message transport service.

This was the environment in which the EIN project began life, and it learned from the predecessor, adapting the datagram technique and many features of the TELELINES transport protocol by this time the centre of research had moved away from the communication sub-project, which was well understood, towards the protocols needed to support network applications.

3. CRITICAL FACTORS IN NETWORK DESIGN

Packet switching can be regarded as a tool for the economical and economic development of distributed information handling systems. The convenience and economy lies in the whole system, not just the communication component and it stems from the dynamically multiplexed interface, the dynamic flow control and speed change, the success of packet switching lies in the services to the end user.

Some years of experience since the first packet switching systems began to operate have taught us that there are three factors above all, which critically affect the quality of the network's services.

The first is that our ability to design machine interfaces which are convenient and natural for most people to use. The second factor is the reliability and availability of the system. They cannot become an integral part of industry and commerce unless they can be utterly reliable. The third factor is the ability of the system to adapt itself to changes in technology and to user requirements.
Davies D.W. - A decade of development in computer communications

When we look at network services such as electronic mail, teleconferencing, and groups to understand the conclusion that the scale of a good human interface, availability, reliability, and adaptability are best met by large computer networks with heavy central management. This technology will ensure that this intelligence (and the storage capacity) is available both to the main computer and to the terminal. But the need for larger central systems remains. This is the role of the management register, a global, distributed centre for accuracy and authentication, and a "yellow pages" to tell the user where to look for the service he wants. Our proposal is that not only does technology persuade us to build more intelligence into the terminal, but it is also a principle of good system design.

For the near future we can redefine the network as "that part of the system which serves a large majority of users with common information requirements" and the terminal as "that part of the system which serves one user at a time."

4. THE HUMAN INTERFACE

The chief role of intelligence at the terminal will be to transform the human interface of the system. Network systems are notoriously difficult to operate, because with few exceptions, the human interface has been designed by and for computer specialists. The general standard at present is poor but there are some shining exceptions. They show us what can be done, but we have little idea, in a scientific sense, how to achieve it.

Information processing standards are forming, but not even the most optimistic forecast can anticipate a network system in which the user can have a more uniform or better organized than those in existence today. The rate of technology development and the growth of new services will only make this goal more elusive. Consequently the user's own terminal must be designed to help and provide a consistent interface.

The present day user of network services suffers from a lack of continuity in the human interface, even in the most trivial details. Having mastered the basic skill of interacting with one service the user should always find it easy to deal with another quite different service. For example, he should be able to move from an airline reservation service to a bank service without being conscious of a new language. If we cannot have conformity to comprehensive human interface standards at the user level, at least the terminal can transform the user's experience so as to be as consistent as possible. This transformational capability will be a primary function of terminal design in the future.

Intelligence in the terminal can transform an information service in many other ways. It can select and store information for a particular user profile. A high speed data stream from a broadcast satellite can be used to update remote data bases information which is widely used: financial, sports, weather, news, etc. The storage capacity which will be available will allow people to collect the data bases they want to keep and have fully-updated information always available. In this way a broadcast service can be transformed to serve not only as a personal service.

In the same way, the use of knowledge bases obtained by a network can provide special services on demand. These could offer "conversational" service to all kinds, medical, legal, taxation, leisure activities, etc. The cost of access to these programs must be low and the updating frequent enough so that most users would prefer to pay for it again rather than lose it.

Our discussion has focused on the human interface, but we believe that an analysis of reliability, maintainability, and network management mechanism should also show the value of using the simplest mechanisms at the centre and building the complex parts (admission, measurement, user languages, etc.) close to the periphery.

Packet switching, particularly of the datagram kind, is the main method of transport forwards towards the periphery. The communication service is very simple and it is the job of the packet switching network to provide the communication services. The network has the job of reliable delivery of simple packages, but what to do when they are to be interpreted and what to do if there is a temporary failure which all the communication parties. The reliability performance of the network is high, but it has the advantage of recovery to the user equipment, where it must lie, because only here are the goals and requirements of the communication services fully understood. In this way, traffic with different requirements of accuracy, security, security, and rate, can be handled in the same network.

5. THE POWER OF INTEGRATED NETWORK

This picture of a relatively unstructured communication network which provides the terminal with a menu of options, is quite different from the one we are aiming for. The packet switching has changed in many significant ways. What once was a method of packet switching might now be a simple technique for an integrated network carrying all kinds of information by fast circuit switching methods. The social network has moved on from face-to-face to electronic communication.

Two observations should be made in this description. One is the existence of multi-functional plans for public switching in many developed countries, which will result in the situation that while many service are available, the public network is being called rather late for the demand. The complexity of the user's interface to a network and the investment in protocol development, makes a changeover to an "integrated" network at a high cost. The argument is that the integrated network will preserve the L.I.M.I.T interface but replace the old circuit switching methods in order to deal with the cost-conscious design of the telephone system. Furthermore, switching from fast call establishment to an integrated network will contain a high-speed data control-signalling system which could be provided at any point.

A second observation is that an integrated network was so important a goal that the telephone and data networks would have merged already. Their traffic patterns are not to each other but to national and intercontinental data services. Different forms of traffic such as voice or data, will certainly share common sub-systems, such as the main switches, and other traffic patterns. In order to get a different network equipment, it will require a very serious change in the design.

The packet switching services will offer independent packets (so-called datagrams) for the network of the type of logical circuit that is suitable for longer messages. To achieve high availability and reliability, the network will probably use a transparent mechanism (perhaps independent packet, and enabling a logical circuit to be set up and used at the user level). Because the high traffic level and the complex network delays in the main trunk will be quite small, i.e., most at a few milliseconds. The idea of a satellite link will be avoided in the main trunks, with the exception of certain satellite links to be within the scope of the telephone service. The delay will be in local access lines, so delay will be almost independent of distance, like the telephone service.

6. THE END POINT

The ESI project has opened a period of rapid change in network ideas. When it began, packet switching was a controversial new technique and it was opposed by the established telecommunication carriers. As the project ends the era of the public packet switched network begins. In Europe there has been a great need for new communication services and much has been discovered about existing telecommunication networks, both in Europe and America, and there has been a lot of research done in the area of communication services. As they have shown the adaptability of their ideas to future changes in communication techniques.

The project has been a rewarding experiment in European cooperation. It has done more than anything else to show that a coordination of national and protocol expertise in Europe and given us lessons on how to move towards controversial issues that will arise worldwide in network standards.

There is little doubt that the research of the "network to terminal" correspondence to the periphery of a telecommunication system will continue. Perhaps this will change our basic philosophy of protocol design. Certainly it should be used to enable the central network function of multiplexing, switching and transmission to be simplified. As the ESI project is completed there is a need for similar national European research projects to take its place.

REFERENCES

EIN - an example of cooperative research in Europe

D.L.A. Barber
Director, Executive Body, EUROPEAN INFORMATICS NETWORK
or NPL, TEDDINGTON MIDDLESEX

Abstract
The paper traces the evolution of the COST Project 11 - a European Informatics Network - from its inception to the present day. As a prelude to other papers in this special issue, it introduces the participants who have cooperated together to build and operate an international computer network for research purposes, and outlines their individual roles for developing techniques for its application.

The paper then discusses the part played by the Executive Body in providing a focus for the project as a whole, and concludes by considering some of the lessons that may be learned from this unique international venture.

1. INTRODUCTION

For me, COST Project 11 began on the 15th December 1972 when, as a UK delegate, I attended a meeting in Brussels to discuss a proposal to establish a pilot Informatics Network. The proposal was one of several that had been made by the European Communities (CEPT, Committee Scientific and Technical Research Policy), which had not under the chairmanship of Mr. Aigrain in 1966. These ALP.AR proposals were taken up by the COST Group (European Cooperation in the Field of Scientific and Technical Research) during 1969, and a number of study groups were formed to examine the proposals in more detail. It was the initial meeting of one of these study groups that I attended in late 1970. The study group met for a further four times and mid 1971 had prepared a report which was strongly in favour of the establishment of an Informatics Project, based on the construction of a packet building communication network and the on-load of a research program to explore its applications. An application document (AC 1) about the bringing such a project into being was formulated, and was signed by nine European Governments, together with Buraton, on 23rd November 1971.

Subsequently, two other countries joined the project and so the present signatories are - France, Germany, Italy, Netherlands, Norway, Portugal, Sweden, Switzerland, the United Kingdom, Yugoslavia and West Germany. The Agreement states that the project shall facilitate research into data processing problems, shall permit the sharing of resources having common problems, shall carry on research into the exchange of ideas and the coordination of research programs, shall facilitate the comparison of ideas for national networks, shall promote the agreement of standards, and shall be a model for future networks whether for commercial or other purposes.

A technical annex to the Agreement estimated that the project would last five years. In the first two years a communications sub-network would be constructed linking centres nominated by the Signatories. The Computing System at these centres would thus be joined together to form a Computer Network for advanced research, to be conducted over the remaining three years.

However, it is in the nature of research projects that some tasks prove more difficult than anticipated and EIN is no exception. Accordingly, the Management Committee decided to extend the duration of the project, firstly to the end of 1976 and then to the year 1978. This required a rearrangement of the sub-network at the beginning of 1976, because the Swiss centre was obliged to discontinue its operations. The initial and present topologies of the sub networks are shown in figure 1.

2. PROJECT STRUCTURE

The Agreement provided for the establishment of a Management Committee charged with the overall direction of the project, as Executive Body comprising a Director and three Technical Advisory Groups of experts nominated by the Signatories.

The Director reports to the Project's Management Committee, which is made up of representatives of all the Signatories and an observer from CEPT (the European Posts and Telecommunications Conference). The Commission of European Communities has custody of the Common Fund, as the agent of the Management Committee, and also supplies the Secretary of that committee.

The Technical Advisory Group is the principal technical committee in the Project, and has representatives of all Signatories. In addition, a Centre Coordination Group comprises representatives of the Centres and discusses their technical work, while other committees and groups, mainly responsible for technical tasks, are set up and discontinued as necessary.

3. THE PACKET SWITCHING SUBNETWORK

At the time when the project was conceived, there were widely held views on the form that future data networks should take, and indeed, on whether special facilities for data communications would in fact be required, for many doubted whether data traffic would grow significantly for many years.

The idea of packet switching had been proposed in the early 1960s (1, 2, 3) and research work had been carried out in the USA (4) and United Kingdom (5), but the idea of an international packet switching network was not envisaged at that time. Therefore, we now propose that the basis for a new project should be such a network. Fortunately, events have proved the rightness of the decision, because the majority of public data networks now in service, or being commissioned, are based on the packet switching principle.

The development of the original specification for the subnetwork was carried out by experts from the participating centres under the Chairmanship of the Project Director, and the subsequent analysis of tenders was done by another Working Party of experts who were tendered in accordance with the pre-determined marking system. In this way, an independent objective assessment could be made, even though many different countries were involved in the selection of a contractor. As a result, a fixed price contract was awarded in October 1974 to BELL (France) and Logica (UK) as main contractors, with Ferranti (Italy) and ITxM (Switzerland) as sub-contractors. The subsequent monitoring of the contractor's progress, and the development of prototypes was carried out on behalf of the Signatories by the Executive Body, and the subnetwork (7) was handed over to the Centres on schedule in May 1976.

4. THE COMMUNICATIONS SUBNETWORK

The Network Switching Centers (NSCs) wish form the nodes of the subnetwork are based on the Mita 15 from Compaq International (France). Nodes have either 28 or 32 K bits of storage, a visual display unit and an associated printer, a teletypewriter and paper tape peripheral. They also have developed many hardware interfaces for the communications links to other NSCs and to Subscriber Computers. These implement the 150 Baud standard for data transmission that applies a non-cyclic error detection transmission. The interface is capable of operating at 48 K bits per second. The network communication links operate in full duplex at only 9,600 bits per second. They are leased telephone lines, and call the total rental being about $2 M per year.

The subnetwork also has the following functions:

Flow control: congestion is avoided by despatching a packet only if it has a serial number in a window of a certain size, starting from a sequence number of the packet that was last acknowledged.

Transmission error control: errors are avoided by the monitoring of received packets and the automatic requesting of a retransmission when errors are detected by the checksum mechanism.

Adaptive routing: the routing of packets to avoid faulty areas of the network is based on the sequence number of the packet.

Packet sequencing: the ordering of a logical message will be carried out by packets in the network.

Packet containing: the packet will contain the message and be delivered to the destination in the correct order for receipt of the message, and then the message may contain sufficient storage to accumulate and sort the packets itself.

Delivery confirmation: confirmation of the final delivery of the message is available at the request of the originator.
Prevention of diagnostic information on non-delivery, and of "trace" information: information on the route a packet has taken through the network; both these features are useful for monitoring the performance of the network, as well as in developing the systems. They are again optional at the request of the originating system.

Facilities for collecting operating statistics:

Flexible arrangements to respond to a Network Control Centre, including the ability to completely reload the software of an NCC over the network from any other NCC.

Other technical facilities, such as preservation of the originating reference number attached to a packet, and a set of "filtering facilities for the operator" to select various facilities, for example the amount of network time on request, the facility to send back a packet to the originator ("echo"), etc.

A great deal of experience was gained in the design and development of the EIN network and this is well described in literature produced in the past five years. At the time of writing, intense activity is being conducted to gain as much information about the operation of the network as possible, because the time is rapidly approaching when the network will be phased out and the participating centres will transfer their systems to Euronet, the international packet-switching network that will become operational during 1978 [8].

**EIN MAP dd/mm/yy hh:mm**

![EIN MAP](image)

**Figure 2: Subnetwork Status Reports for Users**

### 5. Subnetwork Control

When EIN operations commenced, some faults were found in the software of the NCC; this was only to be expected with such a complicated system which was not then tested and was connected to the Centre. But the majority of the problems were solved by the Summer of 1977. Subsequently, it has usually been possible failures, or other such external events, that have caused corruption of the software serious enough to warrant a complete reloading of the NCC. The EIN hardware also has not proved on the whole reliable, even though the subnetwork consists of only six links, the adaptive routing algorithm helped to overcome some difficulties experienced in the initial phases of use, and it has been proved to react in a stable way to variations in traffic.

The operation of the subnetwork is distributed amongst five organizations, but it has been accepted that the National Network Control Centre (NNCC) is necessary, because a central control that has information from the whole of the network can perform a much more rigorous detection of faults than one which has only limited local information. This conclusion is one valuable result of international cooperation in network operations, and an example of how best at present is that of the best administrative arrangements for controlling a distributed network.

In fact, a minimal network control function is possible from the operator's console of any NCC, but the facilities are not extensive, and the operator needs a considerable knowledge of the details of NCC operation to use them. The NPL has therefore developed an NCC system (9) that operates on a small computer, connected to the NNCC as an additional Subnetwork Control Centre. Its purpose is to monitor the overall performance of the subnetwork, in order to achieve reasonable levels of service and availability, and to reduce operational manpower by taking automatic corrective action wherever possible.

**In particular, it is active in areas related to the development of high level protocols for open networking such as programming structure, verification and application-specific implementations. The group also contributes to national and international standardisation efforts.**

**CICG** - Centre Interuniversitaire de Calcul de Grenoble

**CIGS** - Centre National de Calcul Mathématique (CNRIM)
studies started on the ODE Network and continued in 1972 on the Cyclades Network. Participation in EIN began in 1976. This activity has taken two directions: distributed applications, and methods of connecting various systems (such as computers and terminals) to a network. The first covers mainly the area of Network Job Control Language, and distributed data bases. The second has allowed CINE and INAE to gain expertise in connecting computers to a variety of networks, using several solutions: internal adaptation, front-end processor or microprogrammed black box. Using these techniques, CINE operates a part of the Cyclades network to allow interconnection of its computers and terminals.

CILEA - Consorzio Interuniversitario Lombardo per Elaborazione Automatica

CILEA is the regional centre whose main activity is in supplying computer services to the five universities in Lombardy, including the Politecnico di Milano. It has implemented and manages the SARPA computer network for batch users, which connects UNICAMP, IBM, CIEC and Ingegnerie machines. It is in present pursuing applied research and development in the fields of Data processing terminal concentrators, data bases, and computer system evaluation.

COST - Centro Studi e Applicazioni di Tecnologie Avanzate

Since 1969 COSTA has been active in three areas: education, services and applied research, with the main objectives of transferring technology and know-how to represent the totality of the "mezzogiorno" (such as local government, industries etc.). It operates in fact, primarily in Southern Italy, and its activities are mainly related to information.

COSTA has been working on EIN as a research project, under CEPII's coordination, since 1977. Its main interests in networking are in the development of packet switching problems, distributed data bases (high level protocols) and Centre management problems on network basis.

5. ASSOCIATED CENTRES

Recently some Signatories have nominated Associated Centres, not connected permanently to the network, but capable of access through the public switched telephone network to a number of Primary and Secondary Centres. The Associated Centres that joined in the 1978 presentation were:

OMI - Osterricherische Mathematik und Datenverarbeitung

OMI, founded in 1968 with headquarters near Buns, is a large scale research institute financed by the Federal Republic of Germany and by the State of North Rhine-Westphalia. It pursues application-oriented basic research, applied research, and development in the fields of data processing, large-range basic research, applied research, and development. Projects constitute a unity within OMI that may findings in research are put into practice by means of development projects, which themselves stimulate further areas of research and development. These functions include advisory activities and contract work, in particular for the public sector.

OMI especially studies application-oriented conceptual problems, thus contributing to further development of computer system, their improvement, utilization and application. In this way the research and development activities cover the whole range of hardware, software and applications, and their role in nation, government and society.

In 1973 the former Deutsches Rechenzentrum at Darmstadt became the OMI Institute for Teleprocessing. This is one of the eleven OMI Institutes, and is active in application-oriented research and development in the field of teleprocessing and distributed systems.

Q2 - Stockholm Dataxanxentral for forskning och utbildning

Q2 is the main institution in Stockholm for higher education and research on large computer networks. In addition to its terminal network, it is connected to SCANNER, a Scandinavian packet switched network for distributed databases.

Q2 participates in a computer network project on the problem of interworking a closed user group to a public packet switched network using X.25. The purpose of this project is to get a standard structure of the communication software, and especially a user interface to the human user.

ESS - Kranjska Skupnosti Slovenije

The Information Centre of the Research Community of Slovenia, ESS, was established in 1978 with the aim of providing information on research activity to all interested parties in the country.

Today the Information Centre is working on establishing computer-based information and documentation services in Slovenia in collaboration with other centers specializing in various fields (technological, electrical, medical, agricultural etc). Some applications are being maintained and further developed by the Information Centre itself.

Networks are of special interest due to the fact that on-line data base usage is essential for the efficient information services that are to be provided to the community in the near future. ESS is responsible for coordinating network research in Yugoslavia as far as EIN is concerned.

3. THE ROLE OF THE PARTICIPATING CENTRES

In many cases the experts engaged in the early activities of the project came from the participating centres, although some signatories without centres also provided experts for the various working parties. In parallel with the common activities, the various centres made their own plans for the introduction of the new systems, and so on. In addition, they began to consider the problem of interfacing their own
When the E12 subnetwork was specified in 1977, it was impossible to foresee the details of any future CCITT standard, and indeed it was impossible to predict when a public packet-switching service would be available, so the E12 data link protocol is compatible with X.25 levels 1 and 2, which are based on earlier standards, 10 differs from level 3.

The E1 Management Committee therefore directed the Executive Body to develop an adaptor box, suitable for interconnection between an EIN NSC and the X.25 public network, to allow the changeover to Euronet to take place using permanent virtual circuits, with minimum disturbance to Euronet. This development, known for historical reasons as "box D", was completed early in 1973 using multi-interconnection, and had furnished ideas for several different adaptor boxes, as shown in Figure 5. In particular, "box H" allows Subscriber-Computer service, either directly or via an NSC, to make use of established virtual circuits, and indeed based on the work already done, it is now seen to be possible rapidly to introduce the adaptor box for any of a wide variety of practical requirements in interconnecting computers to a communications subnetwork. The box which will be used to connect with Euronet is known as an EIN-B (EIN Matting Unit-B).

EIN sub-net leased lines, replaced by permanent virtual circuits in X25 network

X25 VC computer systems use EIN subnetwork

EIN DO subscriber computers linked by switched virtual circuits in X25 network

EIN sub-net provides virtual call and circuit facilities

Figure 5 Some possible X25 Adapter Units

11. THE ROLE OF THE EXECUTIVE BODY

The basic tasks of the Executive Body were, of course, laid down in the Agreement. But there were many problems that have arisen during the progress of the project that have demanded action by the Director and his assistants. Examples such as monitoring contracts and the development of EIN have already been mentioned above, as has the part played by the EIN in coordination of Centres' activities. Other tasks have been the representation of the project at public conferences, at CCITT and ISO meetings and in interactions with other projects.

In all these activities, a number of valuable lessons have been learnt about the management of an international cooperative research project, with distributed participants.

For the most part, cooperation between Centres has been satisfactory because, once each particular objective has been agreed, each Centre has been able to work independently to reach its own mission as required. This is in accordance with the provisions of the original agreement.

But with such a complex project a more detailed control of the work is often desirable, because the success of the whole project relies on the proper interfacing of the systems of the individual participants. This has proved hard to achieve through the committee structure adopted for E12.
The achievements of EIN have been generally acknowledged and there is a general good will towards the future of another such project. Some time has elapsed since the research into the application of discrete switching systems, rather than their design, as was the present project. It is too early to see what form a new project would take or indeed if it will be possible. It is important that, given the experience of the present project, the combination of networks that need to be solved if the maximum advantage is to be gained from the integration now going into the new public data networks.

The paper was received on May 11, 1970

REFERENCES


1. Introduction

The European Information Network (EIN) is a computer network resulting from a research project of COST, namely Action 71. The Agreement for the development of this project was signed in 1971 by nine signatories (The Commission of the European Communities, Italy, Germany, France, Switzerland, Spain, the United Kingdom, and later also by the Netherlands and the Federal Republic of Germany). The technical goal of the project was the establishment and development of an integrated information network connecting European Centres all over Europe. The network was to be tested, on one hand, the development and management of a packet-switching communications system, and on the other, the use of the distributed processing system architecture to be implemented in such mainframe connected to the subnetwork (computer network).

According to the Agreement, which was corrected in 1975, a Technical Committee gathering representatives from all the EIN Centres, met to define the functional specifications of the communication network. These specifications were finally agreed and are set out in the following sections.

2. The Future

In the few years since EIN began, significant changes have occurred in the technological environment, brought about, in part, by the development of the project itself. Public data switching networks are becoming commonplace; a growing number of international users has been created by the signature, and Europe is now in the forefront of research in this area. In recent developments, there has been the advent of the microprocessor which is revolutionizing and speeding up the formulation of new applications for information systems. Against this background, the project will grow in size, as the subnetworks are superimposed by the use of Euronet.
This situation has brought to the following considerations:
- No hardware modification was feasible in any SC;
- No modification to any operating system was available;
- The overall network software was running on any SC as a normal user job, often with high running priority when available and necessary;
- The network software was not as portable as possible;
- The network software had to be modular and fail-fast, because unreliable nodes of the software had to be expected, e.g., the most dense area of the implementation,
- To expand services, to add new standards such as the X.25 recommendations, etc.

This approach resulted in the following consequences:

1. The basic functions
   The general architecture of the network software is shown in Fig. 1. The overall system consists of a layer of the various modules, each box in the figure is a group of modules, which will be described in deeper details in Sect. 3.

   a) In (1) (stands for Interlocutor) provides the network access by performing the SC-AG protocols.
   b) The AG-EC is a network protocol that is particularly suitable for the datagram service provided by the network communication network. For allowing the access to the network, it is possible to use normal STP and other public data networks. It is possible to use also the standard X.25 interface.
   c) The group (TS stands for Transport Station) performs the EDN (End-to-End) protocol and provides services on the top of the EE protocol, such as echo, delay, access to the network for local users, access to local computing facilities for remote network users, etc.
   d) The group (SDN stands for Subnetwork Control Module) is a tool particularly suitable for experimentation and learning on the network. The group can generate sequences of packets and is allowed to install all the facilities normally accessible to the local user (the T3 is only allowed to use EUGENIO Diagnostic, Delivery Confirmation, Packet Ordering).

2. The main modules
   The main modules are the following:
   - The network connects the main modules by performing the SC-AG interfaces:
   - The main modules are the following ones:

Fig. 1: the overall architecture

Fig. 2: the line protocol adapters
The architecture of the software implemented by CRIE for EIN

As regards the local users, BS allows them to initiate the connection to remote terminals across the network or to get their jobs processed at remote sites (if, of course, they are connected through the network). As to the remote users, BS instead provides them with a way to submit their jobs on the system and to get back the printouts as they were received. BS deals with the transfer of large amounts of sequential data through the network without any delays and accomplishes this using the print server transfer facility.

VI (VT) stands for Virtual Terminal: Terminal (VT) is a type of mechanism used in the Virtual Terminal Protocol (VTDP) and allows the terminal to communicate with any remote application for terminal regardless of the kind of application and protocols concerned with.

VTDP (Virtual Terminal) protocol permits the virtual terminal (VTDP) and other terminal interfaces to communicate with each other. VTDP allows applications to send data to terminals and packet sources in the network.

VTDP is an entity to which a network administrator is assigned. It is used for debugging new network modules without compromising the stability or efficiency of the network software, which is non-modular in a different (logical) sense address associated with the logical SC.

In fig. 1, the lines connecting the different SCs are connected to one another through an exchange of information or to process a flow of command from one module to another; however, the next must ensure that the modules themselves are properly connected. In this case, each module is connected to the other through a specific interface (see fig. 2). The interface is implemented as a set of commands, protocols, and protocols. The interface is implemented as a set of two command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces. In this case, the interface is implemented as a set of command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces.

In fig. 3, the lines connecting the different SCs are connected to one another through an exchange of information or to process a flow of command from one module to another; however, the next must ensure that the modules themselves are properly connected. In this case, each module is connected to the other through a specific interface (see fig. 2). The interface is implemented as a set of commands, protocols, and protocols. The interface is implemented as a set of two command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces. In this case, the interface is implemented as a set of command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces.

In fig. 4, the lines connecting the different SCs are connected to one another through an exchange of information or to process a flow of command from one module to another; however, the next must ensure that the modules themselves are properly connected. In this case, each module is connected to the other through a specific interface (see fig. 2). The interface is implemented as a set of commands, protocols, and protocols. The interface is implemented as a set of two command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces. In this case, the interface is implemented as a set of command sets: one for the module itself and another for the network. The commands are implemented as a set of operations and protocols, and the protocols are implemented as a set of operations and interfaces.
always for either cause, these primitives are
the users to telnet with T5 in the same
way. Of course, some limitations, typi-
cally of the external) must hold to be kept
for homogeneity also for the internal users
(1). The most important of these limitations
is that, upon completion of a command, T5-IIF
cannot service an external user when
interfacing with the operation or interface in any other
way than by changing the content of a shared
area (either in core of an in mass storage).
It follows that, after issuing a command to T5
via the primitive RUTLOG, the user must
done by checking on its completion (via
the primitive RUTLOG) and take the most
appropriate action in function of the command's
status; typically, while the command is "in
action" the user must schedule to check it
gain later on.

- DROP and EOD(6) implement in the top of both
the lettergram and the liaison services as
a tickler that receive letters from remote
sites and respectively discard them or source
them back to the sender. These services are
used for test purposes as well as for checking
on the availability of the T5 level for
remote SCS's.

- OPA (Operator Facility)(7) allows the operators
of the SCS's to exchange letters among
themselves.

- SA (System Access) allows any local terminals
to dispatch - and to receive - letters
through the network, either in lettergram or
in lettergram mode, in order to access
remote services.

In this way many valuable services have been
available nationwide while the network was
still in its testing phase.

- CT (Command Interpreter) allows experimenters
at local terminals to issue commands to T5
and to get back information on their status.
This has been particularly useful for
debugging T5 and T5-IIF.

The module is essentially an improved
interface to the human user: it takes command
lines from the experimenter, which are
written in an easy-to-use language, checks
on their formal correctness and translates
them into the proper commands to T5-IIF; con-
versely, the answers from T5-IIF are printed
out in a way that makes them easy to
understand.

(5) Of course, when necessary to improve the
efficiency of the service, a special purpose
interfacing to T5” is developed for internal
users: this is actually being done to-
together with the development of the second
release of the SCS.

(6) Drop, Echo and Operator Facility have been
agreed by the EIN technical groups as basic
services to be provided by every implementation
just as they were built-in facilities of
T5.

(7) The BRIDGE can be used, e.g., to reach
via EIN the ESA Documentation Service in Pescara
(Roma) through any Italian SCS's.

- TSS connects T5 with the TTY emulator
and allows remote users to have access via the
network (in both lettergram and
lettergram mode) to the TTY emulator and
through this to all remote services that are not
directly connected to the network.

- NSP (Network Synchronization) synchronizes
all remote (echo) services both in
lettergram and in lettergram mode. All
events are recorded on a file, which can be
processed offline to produce reports in
particular at request, e.g., the file
is so systematically processed to produce
daily and monthly reports on the availability
of the remote centers at the T5 level.

An online information service is also
camed by this module when solicited
with an appropriate lettergram, it returns
"interrogation - present on the availability
of every T5, as it stems from the
system scan (fig. 4).

3.3 The SCS group

The "SCS group" (9) is basically a task to perform
experiments on the subnet and control its
behavior. It is constituted of the following
members (fig. 3):

- BCP (B Sentrogram Control Protocol) generates
event reports to the experimental artificial
traffic in the form of sequences of packets
with predefined features as regards,
e.g., IP addresses of local nodes, time
timestamps, in-depth characteristics of
the conditions, etc. It is an "all-purpose"
diagnostic function, whose features
are expected to be updated as appropriate
and varied according to the user's
function and frequency distribution
selected by the user.

- ECM (Environmental Conditions) communicates
information on the actual state of
the environment, e.g., temperature,
humidity, etc., which might affect
the network performance.

- DFT (Data File) stores all the data
generated by the network and its
components, enabling an extensive
analysis of the network's behavior.

- CMC (Control Module) acts as a
network manager, coordinating the
activities of the other members
of the group.

- T5 (Transport Service) connects
the different SCS's, allowing
experimenters to access remote
services.

- T5-IIF (Transport Service with
Interfaceto the Internetwork).

Legend:

Legend:

- T5 (Transport Service) connects
the different SCS's, allowing
experimenters to access remote
services.

- T5-IIF (Transport Service with
Interfaceto the Internetwork).

Legend:

Legend:
The architecture of the software implemented by CREI for EIN

3.4 The Laboratory (LAB)
The LAB is an entity that is used for developing or testing any part of the software without disturbing the operations of the other areas for simulating special situations or testing programs, and so on. It is defined as a logical SC, the EN node. In fact, it has four ports for four physical connections to SCs (physical SCs), however, several logical SCs can be defined at the same time, i.e., in the same physical SC. The maximum number of logical SCs in the node managed by CREI supports 9 nodes, 20, 24 (presently assigned to the MUVAC mainframe), 35, 39 (presently assigned to the EBRmainframe), 40, and 41 (frequently assigned to the MUNIV mainframe).

Let us now suppose that 15 and 20 are operating at addresses 32 and 33, respectively, and that a new service, e.g., the remote service, has to be tested before its integration with the rest of the network. The node being active at address 32. This will be done by defining a new 15 and 20 and placing 24 completely at the disposal of people working on the remote service. The normal operation of the network software being active at addresses 32 and 55 is not disturbed.

Analogously, address 34 has been actually used in parallel with addresses 32 (LAB and service) and 35 (CREI) to run an executive program for special tests on the mainframe AT/17S. The same address has been used, e.g., for the TYTe controller at the ENI examination at ENIC (Venice) to connect to the EBR ESA documentation service.

5.5 The Input/Output Controller (I/O/C)
The I/O/C allows a bidirectional exchange of messages (I.e., lines of characters) between its port and a mainframe terminal, defined at generation time. It can command I/O/C to establish connections between any pair of the following elements:

- Software Concomitants
- Terminals
- Packet sources in the network: they address their messages to any SC, which in turn forwards them to I/O/C. These messages are processed by I/O/C as if they came from local terminals.

In this way a number of terminals in the network can use the software modules in any SC, or any other nodes in the physical location, and in a logical SCs as long as the node is not integrated into the network. The I/O/C provides two operational modes: connected and transparent. In connected mode, the remote process can be routed to the module in the other node, and vice versa. In transparent mode, the messages between nodes can be established through the terminal interconnection.

S/C

**Fig. 5:** The SCN group structure

**Fig. 6:** The Virtual Terminal structure

**Fig. 7:** The Batch Station structure
The architecture of the software implemented by DRII for EIN

consists of a symmetric exchange of attention among the remote users. The architecture is depicted in fig. 1. The basic software modules are:

- The User Unit (U) that interfaces the local users’ workstations and the EIN network;
- The Control Unit (C) that manages the remote users’ connections;
- The Interface Unit (I) that provides the User and Control units.

The User Unit performs the Virtual Terminal Protocol and interfaces the Transport Service (TS) at the immediately lower layer.

The proposed method for communication among EIN’s remote users is a simplified version of the bulk transfer protocol proposed with the addition of queuing and message storage at the server. The protocol is described in a previous section. Details of the protocol and its implementation will be discussed in a future paper.

On the contrary, the architecture of remote users’ jobs as implemented for EIN local users, the systems, and the control policies of the systems, are different and require specific solutions. The systems design is based on a server-client model where the server performs the bulk of the work, and the client provides the user interface.

5. IMPLEMENTATION AND TESTING REQUIREMENTS

The implementation of the software requires careful consideration of the interactions between the different modules. The software is designed to be modular, allowing for easy testing and debugging. The design also ensures that the software can be scaled to accommodate an increasing number of users and connections.

In the prototype implementation, the software is tested in a controlled environment. The performance of the software is evaluated by measuring the time taken to complete tasks and the network bandwidth used.

The software is designed to be scalable, allowing for easy expansion as the number of users and connections increases. The software is also designed to be robust, ensuring that it can handle unexpected errors and failures.

Table 5.1 lists the modules and their implementation details. The table includes the module name, its implementation details, and any additional notes.

4. COLLECTION OF STATISTICS

In a research project like EIN, the collection of statistics on the various aspects of the system is crucial. The following section describes the collection of statistics and the software used to collect them.

The collection of statistics is an essential part of the system’s evaluation. The data collected helps in understanding the system’s performance and identifying areas for improvement.

Table 5.1 lists the statistics collected and the software used to collect them. The table includes the statistic name, its description, and the software used to collect it.

The software used to collect statistics includes the following:

Table 5.1 (Continued)

The software used to collect statistics includes the following:

Table 5.1 (Continued)

The software used to collect statistics includes the following:

Table 5.1 (Continued)

The architecture of the software implemented by CREE for EIN

### TABLE 5.3

<table>
<thead>
<tr>
<th>MODULO</th>
<th>CODE</th>
<th>DATA</th>
<th>NOTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>E1</td>
<td>1367</td>
<td>2224</td>
<td>2357</td>
</tr>
<tr>
<td>E1</td>
<td>10818</td>
<td>2462</td>
<td>5501</td>
</tr>
</tbody>
</table>

**100 ports, 25 distant stations (15 distant and 5 near stations available at each port)**

| E2     | 71   | 37   |       |
| ECHO   | 96   | 28   |       |
| ECHO LIGION | 103 | 33   | 5801 | 90 |
| SF     | 2126 | 756  | 1548  |
| IF     | 2195 | 1004 | 2320  |
| IF - TS | 1553 | 940  | 1955  |
| IF - T | 7101 | 1635 |       |
| UTILITIES | 3692 | 3943 | 1635  |
| SCR    | 7650 | 2380 | 3459  |
| VT     | 4797 | 1127 | 8280  |
| NS     | 4923 | 5256 | 1410  |
| T/0/C  | 2227 | 1516 | 1444  |
| FTV EMULATOR | 921 | 1310 | 50    |

### TABLE 5.4

<table>
<thead>
<tr>
<th>CPU</th>
<th>110E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Banks</td>
<td>512 Kwords each 3 ns across time</td>
</tr>
<tr>
<td>1</td>
<td>8044 Disk Drives 95 ms access time</td>
</tr>
<tr>
<td>2</td>
<td>8044 Disk Drives 35 ns access time</td>
</tr>
<tr>
<td>Operating System</td>
<td>EXEC-0 V33 RS</td>
</tr>
</tbody>
</table>

The system here described is the result of the common work of all the authors, however, a majority of them, namely A. Bellini, M. Degrazia and R. Gerosi, have rendered their names against the task of editing this paper.

**APPENDIX**

In 1974 an Agreement was signed by the Ministry of the Scientific and Technological Research and the Government of Italy, France, Switzerland, United Kingdom, Yugoslavia, Norway, Portugal, Sweden and, more recently, the Netherlands and the Federal Republic of Germany. This Agreement started the Association of the Activity II (known as CREST II) of the Program of Cost (Commission Europea de Coopera fioni scientifiche e tecnologiche) concerning the design, the implementation and the experimentation of the European Information Network (EIN).

The first five of the eleven Signatories mentioned above took an active part in the project and more recently their five EIN nodes are: in France (Magno), Milan (CREE Italia), Paris (IDATE), Turin (CRI) and London (NPL). The other six Signatories took part in the project without installing nodes. However, they can install nodes at any time.

In Italy, the Ministry of the Scientific and Technological Research has been charged with the task of following the whole Italian part of the project to this purpose, and an IC has been created in 1976 as a result of a convention between the Ministry and the Politecnico di Milano. The name of this centre is I.C.R.E.T. (Centro Rete Europeo di Informatica E.T., CREE follows all the aspects of the project and of its experimentation, from the technical ones to, according to the guidelines stated by the Ministry, the managerial and political ones.

The COST II project has some aims that are clearly stated in the Agreement and can be summed up by saying that the project has been largely useful in reducing the time required for the implementation of new applications of the kind already developed or, more generally, the experience gained in the implementation of the system's programming, at the upper levels, TS Services and IF, largely beneficial of the experience gained in the implementation of DS. It is also obvious that the global efficiency is increasing and that the analysis of most of the TS Services has been very fruitful.

**21. T/0/C and TTY-EM, in the end, have been written directly in Assembler. These two moduli contain**

### TABLE 5.5

<table>
<thead>
<tr>
<th>MODULO</th>
<th>NS/MONTH</th>
<th>MODULI/MONTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>TS</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>12</td>
<td>0.78</td>
</tr>
<tr>
<td>UTILITIES</td>
<td>12</td>
<td>0.63</td>
</tr>
<tr>
<td>VT</td>
<td>12</td>
<td>0.42</td>
</tr>
<tr>
<td>T/0/C</td>
<td>6</td>
<td>0.42</td>
</tr>
<tr>
<td>TTY-EM</td>
<td>3</td>
<td>0.30</td>
</tr>
</tbody>
</table>

**TOTAL 106**

Vol. LVIII - 6 AGOSTO 1979
The architecture of the software Implemented by DREI for EIN

- Consiglio Nazionale delle Ricerche (CNR)
- C.N.1.
- Assoscatio Nazionale Industria Elettroniche and Automazioni (ANIE)

The paper was received on May 2, 1976.

REFERENCES

/14/ G. Le Moi: Implementing the Software for the Carter of a Subnetwork - IEEE 78 Toronto, August 3-6, 1976.

Gateway on higher level protocols

P. Schicker
BELL NORTHERN RESEARCH, OTTAWA
A. Duenki
UNION BANK OF SWITZERLAND, ZURICH

ABSTRACT

The design of the virtual terminal gateway in Zurich, mapping the between the Zurich and COST virtual terminal protocols, is presented, with an ensuing discussion on the experiences and insights gained in this area of network research.

1. INTRODUCTION

The European Informatics Network (EIN), originally known as COST Project 11, is a research community established to an international agreement. The main area of research is the definition and experimentation with network services and their protocols. After establishment of a data subnetwork linking several research institutes in Europe, the first achievement was the common definition of a transport service and its protocol.

Parallel to the implementation of this transport service the Center Coordination Unit (CCU) set up a working party to define a virtual terminal protocol. This working party finalized a proposal by the end of 1976 [2]. Several centers, not satisfied with the achieved definition, set out to continue work in a smaller working group. This activity led to the definition of a second protocol, the Zurich virtual terminal protocol [1].

The European Informatics Network was then in the unfortunate situation that there existed two different protocols for the same kind of service with a number of centers, implementing only one of the two protocols. Although this situation is not without benefit in a research environment where the two different approaches can be compared and experimented with in parallel, it left the research community unable to mutually communicate on the virtual terminal level.

In early 1978, the center at the Swiss Federal Institute of Technology in Zurich (ETHZ) decided that the attempt to map the two protocols into one another was a worthwhile research project and established a virtual terminal protocol that was compatible with the new standard.

2. VIRTUAL TERMINAL PROTOCOLS

With the growth of the terminal market and the accompanying growth in data communication services it is not uncommon to find applications which are accessed by a wide variety of different terminal types, or terminals which are used to access a wide range of applications. The direct handling of a growing number of terminal types by every application becomes costly.

The virtual terminal is the definition of an abstract standardized terminal. The definition is that of the terminal's capabilities and behaviour which is not expected by an application who communicates with this terminal. While the internal organization of the virtual terminals may vary, the terminal, as seen by the application, is well-defined and behaves consistently with the definition.

The virtual terminal definition anticipates the use of public data communication networks, especially networks of the packet switching type. To make good use of the new facilities, the virtual terminal is defined as a message terminal rather than a character terminal.

The virtual terminal protocol is the definition of the sequence and interrelation of information that is exchanged via the data communication service between the terminal and the application.
3. MAIN DIFFERENCES BETWEEN THE TWO TERMINAL DEFINITIONS

The main differences between the two virtual terminal definitions are the following:

- The CCG virtual terminal defines a scrollable node terminal (roughly equivalent to a Telotype) where the Zurich virtual terminal considers parameters of type not worthy of special consideration especially as the virtual terminal in the Zurich virtual terminal considers the terminal of the data entry type (roughly equivalent to e.g. an IBM 3270) for each terminal.

- The CCG virtual terminal allows for all kinds of interrupts (extra details are explained below when discussing the protocol differences).

- The Zurich virtual terminal provides for the possibility to transmit and receive function ordinals (invoked e.g. by pressing a function key). The CCG virtual terminal a concept a concept in this generality does not exist. However, the end of a message is qualified with one out of two codes.

The different terminal definitions are reflected in the protocols, as are the designers' different tastes. The main difference between the two virtual terminal protocols are as follows:

- The CCG virtual terminal protocol is heavily based on the EIN transport stack: it assumes an exchange of blocks only. This is manifested in the definition of the protocol which allows the exchange of blocks only. Every block carries a block header with some control information (e.g. which virtual terminal protocol is used and a kind of status). The Zurich virtual terminal protocol makes no neccessary use of the structure of an underlying transport service and could be implemented as well on top of a transport service that provides for the transmission of a stream of information.

- The negotiation for the setting of parameters in the virtual terminal takes a different approach. The CCG virtual terminal protocol assumes a communication between two limited sets of parameters that the user has to negotiate on the application. The Zurich virtual terminal protocol on the other hand, the communication not only between terminal and application but also between pairs of terminals or applications. The nature of the parameters are different insofar as the CCG virtual terminal allows the setting of parameters such as "Backspace" whereas the Zurich virtual terminal considers parameters of type not worthy of special consideration especially as the virtual terminal in the Zurich virtual terminal considers the terminal of the data entry type of considerable intelligence to cope with the message aspect of the virtual terminal in the CCG termial.

- The Zurich virtual terminal dialogue is organized in four different phases: the negotiation phase, the terminal phase, the terminal phase and the end phase. The CCG virtual terminal knows only two phases: the negotiation phase of the virtual terminal. The Zurich virtual terminal includes an extra phase. The clear-area-phase, that is not precisely defined of interrupt sequenc- e, either clear the end phase. Clear-area-phase or the clear-end of the dialogue.

- The CCG virtual terminal protocol defines three different types of interrupt: asynchronous signals (on interrupt channel), the clear mechanism (synchronized on interrupt and data channel), and the clear mechanism (synchronized on interrupt and data channel). The clear mechanism is the result of the two (on interrupt and data channel) of the EIN transport stack are realized by telegrams and letters).

Fortunately, there are many similarities between the two virtual terminals. Without those a gateway between the two protocols would not have been possible.

4. THE MAPPING

This section assumes that the reader is familiar with the two virtual terminal protocols. We suggest that readers not familiar with the protocols skip directly to sections 5 and 6 where we discuss the transmission of information addressing protocols.

The gateway is so designed that each assistance is given to provide an overview as possible. Thus, for example, the gateway interprets most of the virtual terminal protocol as a virtual terminal protocol and provides the capability of a virtual terminal protocol. The gateway accepts these restrictions in its negotiations with the CCG virtual terminal or application for which it (the gateway) can compensate.

Such intervention is possible only where the two protocols differ in a significant manner. A significant difference between elements of the two protocols, that is, an unambiguous translation can be defined. The gateway cannot compensate for differences between the two protocols. The Zurich virtual terminal protocol and the communication not only between terminal and application but also between pairs of terminals or applications. The nature of the parameters are different insofar as the CCG virtual terminal allows the setting of parameters such as "Backspace" whereas the Zurich virtual terminal considers parameters of type not worthy of special consideration especially as the virtual terminal in the Zurich virtual terminal considers the terminal of the data entry type of considerable intelligence to cope with the message aspect of the virtual terminal in the CCG termial.

4.1. Negotiation Phase

Because the CCG virtual terminal protocol is non-symmetric, an assumption must be made that every time the CCG virtual terminal or application (entity) wishes to communicate to a Zurich virtual terminal, a call from a Zurich entity to a CCG entity is a call from a CCG entity to a Zurich entity. This assumption affects the use of the negotiation phase. The CCG application cannot establish a communication through a Zurich application. A Zurich virtual terminal or application establishes communication through the gateway to a CCG virtual terminal. In the other hand, a CCG virtual terminal can set up a communication to a Zurich virtual terminal or a Zurich application can set up a communication to a CCG application through the gateway.

Negotiation through the gateway will be successful except where:

- one process does not offer a scroll mode.
- the gateway maps only one mode and forces the negotiations to this option.
- the CCG protocol demands unlimited values for the x co-ordinate. This concept is not available in the Zurich protocol.

The gateway ignores the overprint option of the CCG protocol.

4.2. Data Phase

Surrich CCG

<table>
<thead>
<tr>
<th>Text</th>
<th>Positioning</th>
<th>Text Positioning [see note 1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>F(1)…F(15)</td>
<td>EOM(1) [see note 2]</td>
<td></td>
</tr>
<tr>
<td>📍-CATTH</td>
<td>TRANSPARENT</td>
<td></td>
</tr>
</tbody>
</table>

Note 1: A clear area is used to achieve the desired positioning. E.g., if a CCG process wishes to negotiate a CK but no position is accepted in its Zurich process which requires a backwards position, the Zurich process can be translated by the gateway into a CA and a differential position.

4.3. Attention Phase

Surrich CCG

| A(6) | CLEAR |
| A(1) | A(1) |
| A(3)…A(7) | [see note 5] |
| PURGE | CLEAR |

Note 1: For all synchronized attentions, that is, all Zurich attentions and the CCG clear, the gateway purges incoming information until the synchronizing part is encountered. This keeps with the clear notion that an interpretation which an application may put on the clear will be free from problems introduced by the dual pipelines.

Note 2: A CCG initiated attention which results in an attention being forwarded from the gateway to the Zurich entity and of course, evoke a response attention from the Zurich entity at least as far as the gateway. Response ordinals of A(1) and A(2) are forwarded to the CCG entity. Response ordinals of A(3) and PURGE will, if the original CCG attention was not a CLEAR, cause the gateway to initiate a CLEAR scenario with the CCG entity.

Note 5: Resume and Please are translated and forwarded to the CCG if the mode is alternate and the turn is not with the requestor.

5. ADDRESSING THROUGH THE GATEWAY

The gateway mapping between the two virtual terminal protocols is in intermediate service transport and service, i.e. negotiations are never done with exchange of blocks only. Every block contains a block header with some control information (e.g. which virtual terminal protocol is used and a kind of status). The gateway accepts these restrictions in its negotiations with the CCG virtual terminal or application for which it (the gateway) can compensate.
The solution adopted is to provide at the site of the gateway a range of network addresses each specifying the gateway. Associated with each of these network addresses is a service network address to which the gateway will establish a connection. However, it is called via the corresponding network address. The address associated with a network address is known that a user who is, for example employing a CCG virtual terminal and wishing to access a service network a virtual network terminal needs to know only the specific network address at the gateway site in order to access the service. On the other hand, the resolution is very easy and static and services not forseen in the gateway's tables can not be accessed through the gateway.

To overcome this static and rather unflexible addressing problem the gateway manages a table of additional address pairs. Anybody in the network can update this table by sending a lettergram to a particular address. This lettergram must specify a source address (i.e. the address of the virtual terminal which wants to establish a dialogue through the gateway) and a destination address (specifying the service to be reached through the gateway). A subsequent establishment of a communication (liaison) between the source address and the gateway site address mentioned above will cause the gateway to establish the other conversation to the specified destination address.

6. EXPERIENCE

This gateway mapping two different virtual terminal protocols into each other has only been possible because the two protocols are providing similar services. In addition, the very close resemblance of the two protocol structures has facilitated the implementation of the gateway easier.

The gateway was implemented at the Swiss Federal Institute of Technology in Zurich (ETH). But, unfortunately, the ETH ceased operation as a center on the European Informatics Network (EIN) before the gateway could be used as a routine service by other institutions. It was implemented on a PDP-11/40 and ran under the UNIX operating system [5]. The core requirements were about 28800 (decimal) words of code with each instance of the gateway requiring an additional block of 256 words for variable and buffer storage.

The most difficulties in implementing the gateway are found in mapping on interrupt and attention mechanisms. The very different treatments of this concept in the two virtual terminals and their protocols caused some serious problems for the mapping of the two protocols. In addition, the proper handling of the termination of one of these two virtual terminals also needed some careful analysis. Nevertheless, and another center's tests suggested that we had provided an adequate and usable mapping between the two protocols.

Incidentally, this gateway is the second virtual terminal protocol mapping which we have implemented: the Zurich virtual terminal protocol [4], a logical mapping between the VENUS terminal protocol (VENUS is the terminal protocol and the machine code for which the PDP-11 is the network front-end [7]). It is interesting to observe that the problems of this first virtual terminal protocol mapping implementation also lay in the use of interrupts and termination.

We consider these mapping experiences more than worth their effort as they gave us a very deep understanding of the nature and interrelation of protocols. As a side effect, we also learned that the current state of the art of protocol description (English prose) is far from being the optimum and ways must be found in the future to define protocols in a more formal way.

The paper was received on January 22, 1979.

REFERENCES

1. The Virtual Terminal Definition, P. Schlicht & A. Duenki, Computer Networks, vol 2 no 4, December 1978
2. Proposal for a Scroll Node Virtual Terminal, W. Siegmann & P. Schlicht, Editors, Coordination Group of the European Informatics Network (CCCI), BIM/CCN/7707
7. VENUS, An Interactive Subsystem, P. Schlicht (Dr. Thesis), Swiss Federal Institute of Technology, May 1976

Host-terminal connection techniques in networks

P. J. Winsborrow
Computer Science and System Division, AERE RAEVAL, Oxford

A. K. Duenki
SWISS FEDERAL INSTITUTE OF TECHNOLOGY (ETH) ZURICH

Abstract. There has been considerable interest in the definition and use of virtual terminal protocols within the European Informatics Network community. This has resulted in the provision of a range of network services and terminal access methods. In this paper, the design of terminal concentrators and service access facilities at Benz & Savell in Switzerland are presented and compared. Problems encountered in designing terminal protocol gateways are also discussed.

1. INTRODUCTION

Understandably the great emphasis has been placed on the development of effective tools enabling interworking between terminals and services on computer networks. Such facilities are a prerequisite in a research network like the European Informatics Network (EIN) for communication between establishments pursuing cooperative programs of research. In this environment, some new services for remote terminal access to mainframes is also evident. Within the EIN community considerable effort has been devoted to the development of terminal terminal protocols for these services.

The terminal handling protocol developments within EIN have been based on the virtual terminal approach. In these protocols assume the presence of a reliable transport service [2] providing end-to-end communications functions. They define a device-independent control language for controlling the virtual-terminal services. Although many of the concepts found in present-day terminals are also implemented on the 16-bit data structure with which services interact with terminals. In this paper, we discuss some of the more general interfacing problems which can occur.

2. EIN TERMINAL SYSTEMS

2.1 General

A DEC PDP 11/40 mini connects the EIN-Surich Control Data multimainframe system to the ETH-Surich subnet. This communication processor performs the lower level link protocols, the transport service, interface to the CCI link frame, as well as its terminal handling and virtual terminal activity. These virtual terminal related services are provided on the ETH-Surich end:

- A terminal handling service, which interfaces individual terminals to the network and provides them with virtual terminal services
- A converter service for the central, virtual interactive terminal facility, which interfaces individual users to the EIN facility and provides the conversion between network virtual terminal and local VENUS terminal representations
- A gateway service, which interfaces individual terminal or application facilities of one virtual terminal type (Surich or CCG) with those of the other virtual terminal type.

These services are implemented as subsystems within a specially designed communication system. One process of each subsystem, 'Subsystem Main', is permanent and controls the dynamic creation and destruction of other processes of the subsystem.

While the process configuration of a particular subsystem is special, the process description format itself is identical for any process or process group. The gateway and the converter establish a communication channel with any other processes of the same subsystem for the exchange of data and signals.

2.2 Terminal Handling Subsystem

The physical terminal input/output uses a B.E.C. DVI programmable sixteen-line multiplexer with simultaneously access to up to 9,600 lines/sec. Output uses direct memory access and is individually programmable for each line, but input uses a common 60-character line also.
which requires a software character demultiplexer.

The main process of the terminal handler sub-

system (Main Mail) (CM), can be considered as
"own" this hardware and is dispensed to the memory
processes access rights to particular lines. The

computations carried out are somewhat different, ei-

ther for "Open" requests for some other process, or as a
different event, the generation of a particular
string indicating a desire to use this line.

The request is received by the terminal
processor (TP), which itself is a processor of a particu-
larly high reliability and it serves as a counter in the
appropriate case, e.g., an absolute address for a ccsp with
the same name. It is the responsibility of the tp to
accommodate the appropriate figures from the current
position. For the remaining parts of the exact mapping, attention
between the user processes is referred to [2].

3. HARBELL TERMINAL SYSTEMS

3.1 General

A GEC 4000 minicomputer at Harbell is connected
to the European Information Network (EIN), which
is controlled by the Timeplex controller system. The
connection with a network control package provides
access to EIN. EIN connects to Harbell's minicomputer
via a gateway function between EIN and HESI is supported.

The following virtual terminal services are
available at Harwell:
- a terminal handling service
- a network status display service
- access to an interactive service (SERV) provided
by the IBM 370/165
- access to a PDU-11/45 system connected to the
EIN network.

The terminal handling service, network status
system and SERV access software are implemented
within the Timeplex controller system. The network
status service requires the use of a transport service gateway
in the Timeplex system, the virtual terminal service software
for the PDU-11/45.

3.2 Terminal Handling Service

The terminal handling service is available for
terminals connected to the GEC 4000. This service
provides a mechanism for users to enter and
requests the terminal user to log on, either by
sharing subsystem of the operating system with a
violated virtual terminal, or to log off from an
addressable entities within the transport service gateway
or page terminal operation are supported.

Terminal handling service maintains a process
to provide each terminal involved in network
activity. These processes are created dynamically in
response to user requests. The terminals are connected
through the central system and communicate with its associated
network through the line 4000 terminal. The terminal
handling character level is delegated to the
underlying virtual terminal process, and users
activities are performed through the line service.

The terminal handling service provides
the user with access to a command interpreter and
access to the virtual terminal. The command
interface provided for the terminal
user is controlled by the line-by-line nature of
the interaction with the associated terminal
process. However, this has been used to advantage
in order to design a "comfortable" user interface.

The command interpreter is the central
process within the terminal handling process
work mode in local mode user input is treated as a
request to switch mode and the same
command is used to switch to interactive mode
via the virtual terminal service. It is
possible to switch between "normal" and "interactive"
mode at any time in a virtual terminal session; this
permits a user to read a log at one time and
transmit another at another time. This facility enables
sessions to be made without the responsibility of

503 - 505

ALTA FREQUENCY
rather than the terminal process.

The more important commands available to the terminal are:

(a) Call command:

Request set up of a transport service liaison - BREAK

Request transport process to enter 'end phase' of protocol - ENGLISH

(b) A terminal process.

Request termination of liaison.

(c) Send/Receive operation.

(d) SEND

When X and Y are present the command requests transmission of the virtual terminal page (replaces scroll mode working required) otherwise current parameter settings are returned.

(e) TERMINATION-dismiss.

If a liaison has been established the command requests display of the remote party's address. This is particularly useful if an incorrect address is stored.

(f) POST

Requests display of local transport service address of terminal process. The terminal process address is bound to the identity of the user rather than the terminal.

There are a number of other commands for the help of novice users, the definition of escape characters, the generation of seldom-used protocol features (e.g., CTS board, who control) and the control of a diagnostic tool which maintains a journal of all network activities.

In addition already described there are a number of more frequently used commands which may be of value while the users process remains in normal mode. These in-line commands take the form:

<name><character><command name><parameter>.

In-line processing is useful for editing, cursor control, function input, message termination and attention input. The latter is desirable for use with combine characters as data then the usual conventions of character repetition apply.

3.5 X.25 Network Services

A sub-system is available, which provides network access to X.25 (Harwell uses a terminal). This sub-system permits user access to a general purpose terminal service to access X.25, which is primarily designed for interchanges - like terminals.

The X.25 access sub-system is composed of two interconnected processes, and in the X.25 environment, the information obtained can be accessed over the network by terminals of specific or generic nature (if virtual terminal access is described in 3.3.2). As the application is self-contained the protocols are not large. However, the 3.5.5 protocol does make some assumptions about the integrity of current and display parameters and data sets, which may not always be valid.

4. DISCUSSION

4.1 Terminal Concentrators

Both Harwell and Surch have had little difficulty designing terminal concentrators, although over which they both had total control. The concentrator is basically for general purposes: it is necessary to provide a sophisticated operator interface for control over network failures - for general use it is necessary to provide a sophisticated operator interface for control over network failures - it is necessary to be able to configure the operation interface, the definitions of the network interface which have specific actions, e.g., attention, in a convenient and efficient manner.

5. ACKNOWLEDGMENTS

The authors would like to acknowledge the many helpful comments and criticisms offered during the course of the work by their colleagues in the X.25 community. Particular thanks are due to P. B. Shekdeh and C. C. J. Winters. Need for their active participation in this work. This work was partly funded by the Computing and Electronics Requirements Board of the X.25 Industry.

REFERENCES


The paper was received on December 13, 1979.
The Euronet network: origins, reasons, and possible future applications

M. Mangoni
MINISTERO PT - Direzione Centrale Servizi Telegrafici, ROMA
A. Misino
MINISTERO PT - Assenza Stato Servizi Telefonici, ROMA

ABSTRACT
The article describes the environment in which the issue of implementing the Euronet network was born in the European Economic Community, as well as the environment in which the PTT Administrations of the nine member countries have been entrusted with its realization and the necessary organizational structure set up for this end.

A short description of the Euronet network is given, its possible utilization as a public data network to meet the general data of the PTTs is mentioned and the main objectives already achieved are presented as well.

1. INTRODUCTION
Euronet is an international information network, based on packet switching technology, promoted by the Commission of the European Economic Community which in 1975 assigned the PTT Administrations of the nine member countries with its implementation in order to create a European telecommunication, scientific and socio-economic information network.

Euronet will be put into operation probably in October 1980 and about thirty data banks located in the EEC countries are presently foreseen to be gradually connected; to it according to the CCIIT Recommendation 525.

Euronet has initially been financed by the European Economic Community but afterwards the PTT Administrations decided to contribute the additional necessary funds considering that the PTTs could implement the network also by taking into account their own general aims.

Euronet is potentially in the condition of being upgraded to a public data network.

Euronet and the future of the network in the context of the Community's evolution and in the communitarian management of the EEC.

2. MOTIVATIONS
The need for implementing an adequate data network, enabling the largest category of user terminals of different kinds operating at different speeds.

In order to achieve such an aim, it was necessary to solve the problem of mutual compatibility between the equipment already in commerce by fixing the standards for their cooperation, and it was felt desirable to lay down the characteristics of new kinds of terminals to be taken as reference for the future.

The occasion for a decisive push towards this target appeared when the European Economic Community agreed to the setting up of its own computer network for scientific and technical information.

It was felt that a significant stimulus towards the solution of the above mentioned problems could be derived from entrusting the achievement and management of such a network to the PTTs of the nine member countries, which would thus have the opportunity of integrating and using the network also for their own specific data transmission purposes.

The importance and validity of these points and the idea of looking for a European solution, made their way within the European Economic Community along with the political significance that such an operation would also represent within the context of developing the European Community.

3. SIGNIFICANT STEPS -- ORGANIZATION AND WORK OF THE EUREONET CONSORTIUM
The need for a series of specific initiatives, among which, in particular, the implementation of an adequate computer network within EEC, was indicated, in 1971, in a Ministers Council resolution aimed at coordinating the member countries' activity dealing with scientific and technical information and documentation.

From this starting point the approval was given by the Council for a three-year plan of actions that gave rise, in December 1975, to the stipulation of an Agreement between "the European Community and the PTT Administrations of the nine member countries."

This Agreement, signed by the nine EEC PTTs, grouped in a Consortium by signing a "Convenzione" -- the task of implementing the European telecommunication and information network, the "Euronet" network.

In 1978 the Council of the European Community approved a second three-year plan which would provide for a series of actions in the field of scientific and technical information and documentation to support and develop those actions undertaken under the first plan, and the collaborative plan regarding the overcoming of language barriers among the various countries of Community through the adoption of multi-lingual systems.

The fundamental target is always that of developing, within the EEC, an information network which is more and more complete and accessible to the greatest number of users in all member countries, establishing conditions of equal competition.

The Consortium of the PTTs adopted, from the moment of its constitution, the following organizational structure:

-- Management Committee, with decisional functions; they meet when necessary and each PTT Administration is represented by a member.
-- Technical Planning and Implementation Committee, composed of the technical representatives of the PTT Administrations, with consultative functions, respectively, on technical matters and on administrative and tariff matters; they meet when required and each Administration is represented by a member.
-- Project Team with permanent members coordinated by a Project Director; they work closely with representatives of the Commission to ensure that the network meets with the agreed objectives and is provided within the agreed time table.

At a later stage, in order to solve national installation problems and to face technical and operational natters and equipment maintenance, respectively, a Realization Group and a Maintenance Group were set up in each country.

Lastly, a Commercial Group with promotional and user assistance functions has been set up in each country.

The choice of the network project was made in December 1979.

The project chosen was the one proposed by a group of European Companies led by the French SSEA, which had presented a proposal derived from TRANSIPAC network technology. This project appeared to be more consistent with the view of a possible diffusion and upgrading of Euronet to a public data network.

In June 1979 the French Administration, also on behalf of the other PTTs, signed a contract with SSEA for the provision of both hardware and software.

During the same year the international tender for modems at 28 Kbit/sec between the network nodes was assigned to the Italian Company ITALTEL with a contract signed in February 1979 by the Italian Administration, also on behalf of the other members.

The contract for supply of modems at 9.6 Kbit/sec for connections between the equipment at some remote access points and their serving node was assigned to the French Company SAT.

Due to the complexity of the problems to be solved (most of them are of a new kind and subject to agreement by the Administrations and by the EEC Commission), the activation of the network, which had been foreseen for the end of 1979, will take place in the second half of 1979.
4. A BRIEF DESCRIPTION OF THE NETWORK

The initial topology of the network is shown in figure 1; four nodes (FSE, packet switching exchange) are located in Frankfurt, London, Paris and Rome, and five remote access points, composed of time division multiplexers, are located in Amsterdam, Brussels, Copenhagen, Dublin and Luxembourg.

The four nodes also perform the functions of assembling the characters in packet and vice versa (PAD, packet assembly/disassembly) in order to allow the interchange of traffic between packet and character mode terminals. The network will verify the virtual call and permanent virtual circuit facilities.

It will adopt a fixed routing method, that is, all packets of each call will follow the same route assigned to the first packet of the call.

The advantages over the adaptive routing method (that is, independent for each individual packet) are that the packets do not need to be reordered at the arrival node, and the accounting operations for transit calls are easier.

A network management center (NMC) is located in London; apart from carrying out dynamic working operations on the network, it will provide for the recording of billing and statistical data.

The network, as already mentioned, is a packet switching type derived from the French Transpac network technology.

The inter-node protocol, which at the opening date of the network will not conform to the recent international standards, will at later date presumably be brought up to date to the CCITT protocol X75.

The nodes are the nodal type and they are composed of the following basic parts, as indicated in the simplified block diagram in figure 2:

- group unit (GU); it implements the PAD facility and performs the packet switching function; the capacity of the associated memory may be extended up to 240 K words;
- synchronous line adapters (SLAs); they implement the synchronous packet interfaces; at the most 32 SLAs can be added;
- synchronous line unit (SLU); it carries out control and transmission functions between

---

**Fig. 2 - Euronet node - Block diagram**

**1. CP 50 switch modules:** they are made up of a purpose-designed minicomputer developed in France for Transpac network; they support the connection of users and perform the numerous repetitive functions implied in the switching of packets; at the most 32 modules can be added; each of them can support up to 260 synchronous packet interfaces (synchronous packet ports) and 240 asynchronous-character interfaces (asynchronous ports); the synchronous ones will connect packet mode terminals, including data banks (books), operating in accordance with CCITT Recommendation X25; the asynchronous ones will connect terminals operating in accordance with Recommendation X28; the number of the necessary CP 50 modules depends on the volume of the traffic and the number of the users to be connected.

Each CP 50 module is composed of the following main units:

- the SLAs and the GIU;
- asynchronous line unit (ALU); it carries out control and transmission functions between the asynchronous connections and the GIU; the ALU processes the characters coming from the asynchronous lines so that the GIU presents to the GI an interface similar to that presented by the GIU;

**2. Command unit (CU):** it is made up of a general-purpose minicomputer, the SEMS MITRA 125, whose associated memory capacity is 64 K words; the command unit performs the functions that necessitate a more complex treatment; with its own software it forms the most "intelligent" part of the node; the CU controls the operations of the CP 50 and runs the signalling, the construction and the clearing of the calls; at the most 8 CU modules can be added and their necessary number depends closely upon the number of calls per time unit:
3) CASE 670 and 672 time division multiplexers; the 590 ones implement the asynchronous-character interfaces and connect to the national asynchronous terminals either by point-to-point circuits or through the public switched telephone or data networks, the 672 ones connect the equipment of the remote access points (these are made up of CASE 672, which connect the synchronous packet mode terminals, including data banks, and of CASE 670 which connect the asynchronous terminals).

The present dimension of each node is made up of two command unit (one main and the other stand-by) and one CP 50 module (with double unit). The distribution of the interfaces for the users of the various countries, forecast at the moment, is indicated in the following table:

<table>
<thead>
<tr>
<th>Country</th>
<th>Character mode</th>
<th>Packet mode (included nodes)</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>32</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>Denmark</td>
<td>32</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>German Federal Republic</td>
<td>64</td>
<td>31</td>
<td>95</td>
</tr>
<tr>
<td>France</td>
<td>0 *(c)</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>Ireland</td>
<td>32</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>Italy</td>
<td>48</td>
<td>35</td>
<td>83</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>26</td>
<td>4</td>
<td>26</td>
</tr>
<tr>
<td>Netherlands</td>
<td>92</td>
<td>6</td>
<td>98</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>64</td>
<td>25</td>
<td>89</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>336</strong></td>
<td><strong>122</strong></td>
<td><strong>458</strong></td>
</tr>
</tbody>
</table>

It is foreseen that in the countries other than France the asynchronous terminals will initially gain access to Euronet through the national switched telephone network; a contention ratio of 1:1 for the relevant parts has been assumed. For the terminals that gain access through the telephone network, the singling out of the calls in debiting the traffic carried out, will be operated by means of an identification code (VNI network user identifier) that will be set by the caller and will be controlled by the node on its own internal list of the identification.

*(c)* The character mode terminals will be linked up directly to the existing French Transpace network that will be interconnected to the Euronet one by means of an appropriate international transit node. If also the packet mode terminals will be linked up directly to the Transpace, the eleven interfaces indicated for France will be used as spare parts.

Also the forecast implementation time of the national data networks of the various countries.

5. PROSPECTS FOR EURONET TO BECOME A PUBLIC DATA NETWORK – MAIN RESULTS ALREADY ACHIEVED

As already mentioned, Euronet was conceived as a potential embryo of a public European data network. However, the hypothesis of extending, in the various European countries, the Euronet network to a national level through the addition of Euronet nodes not yet carried out and the various PTT Administrations are now oriented to setting up their own national packet networks, in compliance with international standards (including those defined after the Euronet project had already started), so that these national networks will be able to work with each other directly.

If Euronet is upgraded to all such international standards, it will be able to work with all national packet networks.

This upgrading appears to be very important because only in such conditions will Euronet be able to provide a concrete possibility of operating switching data traffic among all the countries of the Community, independently from the different times of implementation of their respective national networks.

Back up traffic in international relationships which would not justify the setting up of or increase of bilateral direct connections from country to country.

In any case the Euronet initiative has already achieved some important and concrete results, as for example the following:

- a solid base for the development of cooperation among the member countries in matter of scientific and technical documentation;
- an action of adaptation and organization in an area characterized, up to now, by proliferation of non-coordinated initiatives and private networks incompatible with each other;
- a general stimulus for the implementation of national data networks and for packet switching nodes;
- a census on, and a sort of publicity for European information resources;
- a concrete incentive to the constitution of new European data banks and to the diffusion of information to benefit the largest level of users;
- an incentive to the definition of international standards for data transmission networks;
- an approach to the problems of the telecommunication and the computer worlds in unity of scopes and interests and looking for an European inclusion in an area of fundamental importance for every advanced progress;
- a promotional incentive for European industry;
- the definition of rational and homogeneous telecommunication tariff criteria, based on the volume of data transmitted without taking into account the length of the involved international paths.
6. CONCLUSIONS

Euronet has undoubtedly played a fundamental leading role within all the community member countries and has probably promoted in a tangible manner the generalized acceptance of the packet switching technique.

Other European countries (not belonging to the Community), have shown interest in joining the Euronet Consortium. Switzerland has just signed an Agreement to participate in Euronet activities with a European node in Zurich; Spain is waiting for the new protocol X.25 to link up its own packet switching network with Euronet, and Norway and Sweden are studying a suitable solution to be connected to Euronet access points. Also Greece, a new member of the EEC, and presumably Austria are interested in the Euronet connection. Generally speaking, new associations are welcome because on the one hand, they would bring additional traffic due to the new users which will provide a more efficient network utilization, and on the other hand they could add a greater value to the network itself because of the additional information resources made available to all other users.

Until national data networks are implemented all over Europe, Euronet is the only real present possibility of data exchange among all member countries.

Naturally, this network cannot be considered as anything but the beginning of a development process within Europe of an information exchange market, still entirely to be developed, which will require in any case many additional international initiatives.

The paper was revised on July 13, 1978

---

Network management and control in Euronet

A.M. Repichini
SIF SOCIETA' ITALIANA PER L'EMISSIONE TELEFONICA

Abstract: In telecommunication network design, to obtain better performances in terms of user services, network availability and management techniques have been introduced which utilize in switching nodes "intelligent" equipments, based on general purpose or specialized computers. In this context, one of the problems is the control and management of such a system. This problem, already important in experimental private networks, is crucial in public ones, realized and run by PTT Administrations, which must ensure a high level of quality and continuity of service. In this paper, methodologies and techniques used in Euronet project will be shortly described.

1. NETWORK STRUCTURE

Euronet is an international telecommunication network, realized by a Consortium of European PTT Administrations on behalf of the European Community, in order to create a "Common Market" of technical, scientific and social-economic information.

This network is structured on four Packet Switching Exchanges (PSE), each interconnected, and five Remote Access Points (RAP), placed in a way to link the PSEs to a Control Centre for Traffic (CMT), located in Rome and connected to the exchanges PSE through ISDN links.

The network architecture is based on a microcomputer SEPS MITRA 125. The configuration consists of 128 kilobytes memory, operator console, 20 floppy disks, magnetic tape, 3 VDUs, printers and readers and special peripherals for visual alarms and buzzer.

2. CONTROL SYSTEM GENERAL ARCHITECTURE

Network operational facilities can be divided into different categories depending upon whether they are concerned with the quality maintenance of subscriber service or with the network accounting management.

These functions are therefore essentially concerned with processing network control, supervision and management as well as collecting subscriber's billing information. The main notions characterizing the implementation of those functions are:

- high operational reliability;
- automatization in order to reduce human intervention;
- distinction between distributed functions (e.g., data collection on subscribers or modules status and centralized ones (e.g., observation of the whole network status, co-ordination of control and maintenance operations, etc.);
- functional independence of various control system components, that are logically distinct from modules which assure the subscriber service;
- operational simplicity, essentially coming from the availability to network operators of visualization means and powerful control languages;
- modular design of the interconnection service communication protocols, to allow future evolution of the system.

These concepts led to define, at a distributed level, PSE programs for local network management, subscribers accounting, management of virtual circuits, failure detection and system reconfiguration, preventive maintenance, etc.
A formal description of the DTE packet level in the X.25 Recommendation

S. Alfonzetti, S. Casale, A. Faro

ISTITUTO ELETTRONICO E NUCLEARE, UNIVERSITÀ DI CATANIA — CIEI (POLITECNICO DI MILANO)

Abstract: This paper proposes a formal description of the DTE packet level (Level 3) of the X.25 protocol. The formalization is based on the theory of collocations and uses the logical matrix method. Commands exchanged between Level 3 and its upper level are selected. Some uncertainties of the procedure of the DTE level 3 are pointed out and are overcome by introducing suitable hypotheses.

1. INTRODUCTION

In the last years, computer network study has been greatly developed. A computer network is made up of a communication subnetwork to which terminals and computers are connected. Both these users of the subnetwork are called DTE (Data Terminal Equipment). DCE (Data Circuit-terming Equipment) is the interface of the communication subnetwork with the DTE.

The interface between DTE and DCE is the argument of the Recommendation X.75 of the CCITT [2] for terminals operating in the packet mode. The logical description of this interface is given by the recommendation X.75. The logical description is based on the theory of collocations and uses the logical matrix method. Commands are exchanged between Level 3 and its upper level are selected. Some uncertainties of the procedure of the DTE level 3 are pointed out and are overcome by introducing suitable hypotheses.

In this paper we have formally described the DTE level 3 as an "interconductor" of the theory of collocations [2]. The description technique used is based on logical matrices, state vectors and context procedures [1][7][9]. We present a general and non-ambiguity of such a description technique, it has been possible to point out all the procedural questions of the DTE level 3. To formalize the procedure it has been necessary to overcome such questions by introducing suitable hypotheses, some of which according to [2] and [9]. However, the interrelationships between level 3 and its upper level (Level 4) have been defined by introducing a set of commands as requested by the description technique.

Sections 2 gives the reader a brief recall of the "theory of collocations" and of the logical matrix description technique. Section 3 presents the hy-
consequently the description too complex. Therefore it is advisable to split the PU into two parts, in such a way that one processes the possible parameters or the envelopes (or commands) and the other the opcodes [3]-[6].

For this reason PU contains (see Fig. 2):
- input context (IC), which processes the parameters of the input envelopes and commands and
- output context (OC), which adds the parameters to the op-codes coming from PU.

Moreover, to avoid the interrogator PU remains indefinitely in a given state, it is convenient that PU contains internal mechanisms (time out) which emulate input commands or input envelopes (see Fig. 2).

From above, the interrogator procedure is completely specified if one formally describes the various components of PU.

The formal description technique with logical matrices is briefly recalled in the following.

According to this technique, the PU internal states are defined by means of two state variable nets:

- the first is a set of boolean variables which can be modified by the PDU inputs; such a variable is said to be a variable if it depends only on or, or with or, if it is also depends on or.
- the second is a set of variables which can be modified by the PDU but not explicitly utilized within the state variables; these variables are called context variables.

In and OC can utilize the PDU variables while PU cannot directly utilize the context variables; for this reason a set of binary variables has been introduced (5), which is a function of the context variables and which can be directly used by PU apart from the context; these variables have been called global states (5).

The formal description technique with logical matrices is suitable to utilise programming languages (Algol, Pascal, etc.); while PU can be schematically described as follows:

- at the reception of a command (or PDU produces:
- an output message envelope: $g_{i} + h_{i} + s_{i} + e_{i}$
- an output command envelope: $h_{i} + s_{i} + e_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- a state transition: $i \rightarrow i' + \phi_{i} + \psi_{i}$
- where:

\[ g, h, p, e \]

are matrices of logical functions;

\[ b, r, s \]

are vectors of logical functions;

\[ \phi, \psi \]

are logical functions.

(1) Global states $s$ can be defined also independently by $v$; in such cases they take into account secondary states.

VOL.XLVI, N.8 AGOSTO 1978

\[ 341 \text{ E} - 515 \]
ned (1). The DCE will indicate that there is an incoming call by transmitting across the DCE/DCE interface an INCREMENTAL CALL REQUEST packet (1), sect. 3.1.2. If the call is accepted, the called DTE will answer transmitting a CALL ACKNOWLEDGE packet, permitting the data transmission.

B) We suppose that the DCE transfers a CLEAR REQUEST packet if it does not accept the call (1). On receiving this packet,

The DCE may indicate clearing by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), see sect. 3.1.2. The DTE is required to transmit a CLEAR REQUEST packet to the DCE, clearing the connection. The DTE request can only be acknowledged by the DCE, clearing the connection.

The DCE will inform the user that the connection has been cleared by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is cleared, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been cleared.

C) We suppose that the DCE, receiving a DATA packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

D) We suppose that the DCE, receiving a CALL REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

E) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

F) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

G) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

H) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

I) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

J) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

K) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

L) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

M) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

N) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

O) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

P) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

Q) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

R) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

S) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

T) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

U) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

V) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

W) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

X) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

Y) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.

Z) We suppose that the DCE, receiving a CLEAR REQUEST packet containing a data packet, will send a CLEAR REQUEST packet to the DTE.

The DCE will inform the user that the connection has been closed by transmitting across the DTE/DCE interface a CLEAR REQUEST packet (1), sect. 3.1.2. If the connection is closed, the DCE will send a CLEAR REQUEST packet to the DTE, notifying the DTE that the connection has been closed.
command which is not transmitted to FPU do not appear.

...
A formal description of the X.25 packet level in the X.25 Recommendation.
ACKNOWLEDGEMENTS

The authors wish to thank Prof. G. Le Moit for the helpful discussions and suggestions.

The paper was first received on February 24, 1976.

REFERENCES

process-to-process information exchange. The terminal-to-application and job transfers data communication, also provided by RCPBS, would be mapped on this symmetric process-to-process capability.

From the above requirements the following architectural characteristics were derived:

1. Packet-switching, distributed control communication subnetwork.
2. Availability and reliability of communication systems that can be accessed via a generalized access network.
3. Functional layering allowing a variety of physical node configurations in the view of the associated characteristics outlined above, RPL request activities can be grouped into three fundamental areas:

1. Study and implementation of a generalized communication system. This system should emerge from the union of the software sub-systems and hardware components distributed in all the network nodes. These sub-systems and components interact and cooperate through connections represented by data links or virtual attachments.
2. Definition and implementation of an access method to the Communication System, called RPL (RPL Network Access Control), as a necessary step in the broadcasting of the spectrum of applications accessing the network.
3. Implementation of some applications with the aim of obtaining friends access to the network services for improvement of the performance characteristics of RCPBS, thus validating its functional design.

It is worthwhile underscoring that RCPBS was originally designed to meet ITI (International Telecommunications) standards in order to promote, within an operational environment, an Italian experiment in the network area.

2. NETWORK ARCHITECTURE

RCPBS is conceptually divided into three Layers as shown in Fig 1:

1. Common Network Layer

This Layer provides the packet switching capability by using a physical communication medium. It contains a routing strategy and a well defined packet format. The Common Network switches packets that will be delivered in the same order as they were sent, nor does it guarantee that there will be no loss or duplication of packets. Packets can be switched on other nodes such as the use of alternative routes between nodes that can be chosen to optimize performance or other factors such as the use of information derived from the network state. The Common Network is structured into a Common Network and a Communications Sub-Layer. The Common Network provides the physical communications necessary to transport data from one processor to another processor. The Communications Network is made up of full duplex circuits called Connections. These can be point-to-point, synchronous data links or point-to-point channels.

2. Interface Functions Layer

This Layer provides the interface between the Common Network and the Application Layer (the Layer in which the network services are used). The Interface Functions basically provide:

a) the defining and numbering of ports (for Logical Units/LUs) on the Communications System;

b) extemporaneous services such as query and mailing;

c) setting up and closing down of Logical Channels;

d) the dissemination of Logical Channel status and events.

These functions are performed by the Network Service Manager (NSM) (Fig 2). The NSM controls and manages the Logical Channel Protocol (LCP). Once established, Logical Channels are maintained by a NSM, the Session Manager (SM) (Fig 2) which in turn is maintained by the Open Network Control (ONC) (Fig 2). The ONC provides a full duplex packet oriented protocol which is associated in session with another LU in the network.

Fig 2

The ONC has a window oriented protocol that filters out duplicates, puts packets into sequence for the E.140 protocol packet level and reports this to the ELA and provides for pacing. The ELA presents a dialog message to the PKX module. The function of the PKX is to handle packeting of activity, the delivery of Logical Channels and report message loss to the PKX. The design of the Session Manager is intended to optimize the transmission of traffic, the E.140 packeting of activity and the delivery of Logical Channels. The ONC is responsible for ensuring that the Logical Channels do not become inactive, that the Logical Channels do not become unbalanced, that the Logical Channels do not become lost, and that the Logical Channels do not become unavailable.

3. NETWORK CONTROL AND BRAM

Two software components, already defined above, provide the control to each node.
the Network Services Manager (NSM), within the Interface Functions, and the Common Network Manager (CNCM) within the Communication Functions. Each subarea consists of its respective Layer 2 in an equihierarchical way with the corresponding components in all other nodes and represents an address in the network address space (Pranchi [3]). This separation of control gives rise to a logical node configuration that allows for a variety of physical node configurations. The logical node configuration is defined as follows: in a node, there is one and only one Common Network Manager (CNCM), one, several, or even no Network Services Manager (NSM). By distributing the software layer defined by the architecture on one or two processors, the physical node configurations of Fig. 3 are possible.

This type of unit is called LCT (Logical Channel Termination). Before starting any network activity, an Application asks the Communication System for an LU (Logical Unit), using the SRAM macro instruction OPENLU. This allows the Application to communicate with the system and via this component to send and receive messages and/or inquiries (MAIL and INQUIRE macros). When the Application wishes to connect to another, it sends an operation. An Application wishing to be connected by a BEND request must issue an INQUIRE operation. If the Application requested by the BEND exists and has issued an OPENLU, the two Applications become addressable without the intervention of their respective HMs. If the SR (Session Handler) component carries out the in-system information exchange task by executing the SEND, RECEIVE and BREAK macro issued by the Applications. The BEND Logical Channel allows the operation to be specified at a time. SEND is used to send a message, which will be buffered at the receiving side (given sufficient time) until a matching RECEIVE is issued by the Application at this side. Due to the nature of the half-duplex Logical Channel, it is necessary to change direction information a receiver can send and a sender can receive. The sender is in charge of this direction change. A BREAK can send a message (restricted in length to fit within a Common Network packet) to the receiver in the same state as one that is in send state. The basic format of the message is asynchronous, and no special operation is issued by the receiver. The TELTCG is used by an Application to control certain aspects of the Logical Channel. The TELTCG may be used to release a Logical Channel and a Logical Unit.

4. EXPERIMENTAL IMPLEMENTATION

The network configuration on which the RCFCET architecture is operating, as of July '79, is shown in Fig.5.

The following institutions: -IBM Scientific Center in Venice (1975); -SIAM Laboratory of CNR in Milan (1977); -SETI; CNR Central Administration in Rome (1978);

have joined the project as users of the network facilities.

The central processors available at the node locations are the System/370s, model 150 and 168, and IBM System/7 as Front End Processor (or FEP). The Operating System software on the System/370s in VN/370 in some Nodes and OS/VS (25V/HSAP or MV5/JES2) in the others. Connections between these processors are leased telephone lines. The System/370 Channel Attachment Feature (Local Attachment) can be used between a System/7 and a System/370, when these are located back to back. This type of connection allows a faster, parallel transfer of data between processors. The software providing the Communication System services and the BEND access facilities is OS/VS under VN/370 (Funct [4]) and OS/VS under OS/VS (Soft [5] (Complete for Computer Network Subsystem). Under VN/370, OS is the operating system running on a specialized virtual machine. System/7 network software is an independent stand-alone system called

NCS/7 (Network Control System for System/7). It is written in System/7 assembler language and is compatible with MIPS, the IBM standard support. NCS/7 has the functions of the Control, Storage, I/O and Command Manager (SFS) and the System Control Processor (SCP). The System Control Processor in the configuration of Fig. 5 acts as Full node (see Fig. 5a). The Communication System software and BEND are operational under OS/VS, OS/VS and OS/VS. This means that applications running under the control of OS/VS, OS/VS and OS/VS can communicate with each other through Logical Channels by using BEND. At the moments, the implementation of the RCFCET architecture on any improvement on it, is not available on the IBM System/7 multiprocessor.

Because the RCFCET network architecture is fully independent of the hardware and software on which the network is implemented, various computing centers in Italy, using non-IBM computers, expressed their wish to implement RCFCET architecture on their machines. We were, however, reluctant to encourage this, because EUROSET will become operational in 1979 and because of the present situation of the European PFTs with respect to non-IBM networks.

5. RCFCET NETWORK SERVICES

The following services, which are not yet available, are included with priority in the BEIL project objectives:

a) Interactive Terminal Access; b) Remote File Transfer; c) Remote File Access.

5.1 Interactive Terminal Access

For all interconnections of terminals to Interactive Operating Systems through RCFCET, there will be a Terminal Application at the terminal site and a Host Application at the Interconnection system side. In RCFCET, two different approaches were considered to implement the Interactive Terminal Access. The first approach does not require any modification of the present System/7 architecture (System/7 code); the second, however, required some modification of the System/7 code. The first approach, an IBM System/7 Host Application for a System/7 emulator of an IBM System/7 (System 370/E Communication Control Unit). The System/7 thus appears, to the Interactive computer, to be a System/7 with start-stop terminals (Lancerti [4]). The operating System can support terminals, that have a terminal set up on the Operating System itself, so access terminals via the System/7. On the other hand, the Local Terminal and Application residing in the System/7 control program are essentially additions to the System/7 terminal and application programs. Fig. 6 illustrates the logical separation of the various Application and Communication protocols on this system. The Terminal Application is made up of a line driver that maps a real terminal.
into a Virtual Terminal (VT), a network command processor, and a module for communicating with the host Application via the RPNET Logical Channel. The protocol for this communication is the Virtual Terminal Protocol, or VT (Lasseter). The VT/370 Terminal Application could use this protocol to send messages directly to the terminal by providing the terminals supported by VT/370. The VT/370 Application maps the VT/370 Virtual Terminal Protocol into computer significant operations. For this reason, the Application maps the Virtual Terminal Protocol into 370X/370 emulator operations.

5.2 Spool File Exchange

The Spool File Exchange (Bernett) under VM/370 is provided as an extension of the 3000 and TAC console functions of the VM/370 Console. By using this extension, the VM/370 user at the virtual terminal can consider any spool file designation as a virtual machine running under the processor to which it is attached.

Under CLOS, the VM/370 Conversational Monitor System, spool information is not limited to being logically associated with input card deck or output print lines, but a CLOS user can put on the spool areas any information available on his private direct access space. CLOS users have at their disposal a special command, SEND, which is expanded in the equivalent SPPOOL on the console functions. VM users accessing local or remote card input devices can address any spooling device in the network by putting a TAG card in front of their card deck.

Under OS-VS operating systems, the RASFA facilities were extended to allow the exchange of spool files, not only with other OS-65 systems but also with VM/370 Nodes.

A "SEND card put in front of input card decks provides a switching point to be chosen provided by a TAG card under VM, while a "RECEIVE card allows the routing of output spaces in the network. In particular, jobs to be executed under OS-65 are fetched from the network and then executed by the local or RRC machine, while jobs to be executed on different machines of the same way, spool files to be printed or processed may be sent to other machines of the network which can become acquired without OS-65 execution or scheduling.

The end-to-end protocol adopted provides transmission from VM to OS-65 spool format, and vice versa, for card image or print image data.

5.3 Remote File Access

The CLOS user has at his disposal a set of commands by which he can attach to his host network. CLOS allows the user to link machines in virtual terminals, and connect to another host network. When a remote device is attached, by issuing the RASFD command, the RASFA network used just like a local one, i.e., a list of files in that link can be obtained, files may be read and edited.

RPNET has been implemented almost on schedule and in accordance with the initial design (Lasseter). The objectives and requirements formulated by the project partners have largely been met.

The Communication System, RASFA and the following services: Interactive Terminal Access, Remote File Access, and Remote File Access all are operational. Other services currently under development and testing are now being implemented. The present network will be used, either in its current configuration or with certain extensions. In addition, independent projects on RP stream can be developed for the internal use of the project partners organizations.

The main objective for RPNET in terms of network usage, is to turn the RPNET prototype into a network of national or regional significance. The Italian National Research Council (CNR) and the Italian National Research Commission (CNR) constitute, together with the RASFA nucleus for such a network.

RPNET facilitates research work providing cheap and effective computing services. It newly offers a service bureau type of organization for operation as an independent venture in the current computing environment. RPNET can use the RASFA access method in developing new network Applications.

A large amount of data is already exchanged between the network nodes, because the GRB and IPN research centers are collaborating on common projects. Statistics and accounting facilities are also provided. In order to allow researchers to exchange information with any other network, software which allows an interactive RP stream between any other network supporting TCP/IP and an RPNET was developed. Unfortunately, it is not yet clear whether or not this will be done.

Finally, when a user wants to detach the remote device, he will release it by giving a TERE command, and then issue the command DETSER, otherwise he issues DETSER directly. In both cases, the effect is the same.

6. CONCLUSIONS

RPNET has been implemented almost on schedule and in accordance with the initial design (Lasseter). The objectives and requirements formulated by the project partners have largely been met.

The Communication System, RASFA and the following services: Interactive Terminal Access, Remote File Access, and Remote File Access are all operational. Other services currently under development and testing are now being implemented. The present network will be used, either in its current configuration or with certain extensions. In addition, independent projects on RP stream can be developed for the internal use of the project partners organizations.

The main objective for RPNET in terms of network usage, is to turn the RPNET prototype into a network of national or regional significance. The Italian National Research Council (CNR) and the Italian National Research Commission (CNR) constitute, together with the RASFA nucleus for such a network.

RPNET facilitates research work providing cheap and effective computing services. It newly offers a service bureau type of organization for operation as an independent venture in the current computing environment. RPNET can use the RASFA access method in developing new network Applications.

A large amount of data is already exchanged between the network nodes, because the GRB and IPN research centers are collaborating on common projects. Statistics and accounting facilities are also provided. In order to allow researchers to exchange information with any other network, software which allows an interactive RP stream between any other network supporting TCP/IP and an RPNET was developed. Unfortunately, it is not yet clear whether or not this will be done.

Finally, when a user wants to detach the remote device, he will release it by giving a TERE command, and then issue the command DETSER, otherwise he issues DETSER directly. In both cases, the effect is the same.

6. CONCLUSIONS

RPNET has been implemented almost on schedule and in accordance with the initial design (Lasseter). The objectives and requirements formulated by the project partners have largely been met.

The Communication System, RASFA and the following services: Interactive Terminal Access, Remote File Access, and Remote File Access are all operational. Other services currently under development and testing are now being implemented. The present network will be used, either in its current configuration or with certain extensions. In addition, independent projects on RP stream can be developed for the internal use of the project partners organizations.

The main objective for RPNET in terms of network usage, is to turn the RPNET prototype into a network of national or regional significance. The Italian National Research Council (CNR) and the Italian National Research Commission (CNR) constitute, together with the RASFA nucleus for such a network.

RPNET facilitates research work providing cheap and effective computing services. It newly offers a service bureau type of organization for operation as an independent venture in the current computing environment. RPNET can use the RASFA access method in developing new network Applications.

A large amount of data is already exchanged between the network nodes, because the GRB and IPN research centers are collaborating on common projects. Statistics and accounting facilities are also provided. In order to allow researchers to exchange information with any other network, software which allows an interactive RP stream between any other network supporting TCP/IP and an RPNET was developed. Unfortunately, it is not yet clear whether or not this will be done.

Finally, when a user wants to detach the remote device, he will release it by giving a TERE command, and then issue the command DETSER, otherwise he issues DETSER directly. In both cases, the effect is the same.
SARA:
a network between non-homogeneous batch computers

C. Di Filippo
SPERBI ONYAC ITALIA
P. Mapelli, A. Mattasoglio, M. Zagolin
C.I.L.E.A. SEGROAT (Milano)
G. Meloni
POLITECNICO DI MILANO - C.I.L.E.A. SEGROAT

Abstract. In this paper is described a network for the transmission of batch jobs from the terminals of non-homogeneous computers using a Unix computer as node of the network.

1. INTRODUCTION

The first tests to link up two computers were carried out in 1973 between the 1103's of the Computer Centre of Milan's Polytechnic and Rome University. When the Polytechnic's computer was transferred to the S. Giovanni Wilseno area, and the CILEA was set up, uninterrupted tests were resumed in 1975, due to the need to transfer the workload weighing on the Milan State University's 1103 onto the quicker 1108 without having to install a large number of machines. In January 1976, the link-up between the 1108 and the 1103 became operative, enabling the batchload to be transferred to the more powerful computer. The success of this experiment led to the decision to effect a link-up along the same lines with CDC CYBER of the University of Bologna, a machine which had been used by a large number of researchers in the University area. This link-up began as an experiment in the summer of 1976 and became official at the beginning of 1977. In the summer of 1977, tests began on the link-up between the 370/94 of the CNR of Pisa and connection was fully effected in 1978. At the present time, tests are under way so link-up with a Honeywell 6800 of the Computer Centre of the University of Pavia. In this way, an organic system is created for the link-up of non-homogeneous computers intended to sort batch loads according to CILEA equipment user requirements.

2. STRUCTURE OF THE SARA SYSTEM

The problem which the SARA system solved was that of providing all the users of a certain computer network with access to the calculation capacities of widely differing machines without making any hardware/software variations to the interconnected systems. The need to avoid at all costs changes to the operating systems of the interconnected centres led to research for a system which, although simple, was easily adaptable to any computer, and made it possible to develop a computer network at a time, 1975, when the market offered non-homogeneous hardware systems dedicated exclusively to the interconnection of non-homogeneous computers.

This solution, based exclusively on software and developed by the CILEA of Milan, although designed according to the specific requirements of technical scientists users, besides allowing a high degree of generality, offers the possibility of easily managing and controlling a software product resident on a single machine which also acts as a concentrator.

SARA is a system which manages simulated terminals. This means that in the node computers there are software boxes that are for the connected computers standard terminals. These terminals receive the input queue, (Fig. 1) transmit it for processing to the pre-selected computer and obtain the printouts which they insert in the output queue (Fig. 2). The user who wishes to send jobs via SARA simply activates execution of a batch program whose purpose is to inform SARA of the existence of a job which has to be sent.

The general SARA system is divided into three distinct parts:
- the real-time part: K7
- the batch part: KETEA
- the batch part: SHTEA

First of all, the two programs simulate as many hardware terminals recognized as their own by the
computers connected to the concentrator. As well as incorporating real transmission/reception capabilities, these programs perform intelligent-type programs which enable them to recognize the final destination of the printout arriving. In fact these can be assigned to printers normally connected to the concentrator or to other centres connected via SARA.

Side by side with the functions seen above, RT takes care of gathering a series of management data on traffic, continually updating a special historical file, which is a necessary requisite for the network's efficient information system. Essentially, RTETA programs take care of placing in a suitable queue the jobs to be transmitted to the hosts to which they are destined; the information regarding the jobs placed in the queue is transmitted in an area called TACOCO to which RTETA and RT have access.

Fig. 3 shows the logical function of the system. Firstly, we note that the SARA system, in the past relative to the real time program, can be activated at any time in time, since it is the RTETA program's function to accumulate the jobs to be sent. On the other hand, it is easy to check the situation of the jobs to be sent, since it is sufficient to read the TACOCO file: a similar observation is valid if we wish to save the contents of the queue of the jobs to be sent.

**FIG. 3 FUNCTION OF SARA SYSTEM**

**FIG. 4 DIVISION INTO THESE ACTIVITIES**

**FIG. 5 ANALYSIS OF TACOCO**

The activity regarding transmission of the cards or printouts is carried out by the simulator of the terminal connected to the host. As far as sending cards is concerned, it should be noted that RT reads from the disk the next block, if it is to be sent, prepared beforehand by RTETA and complete in all their parts, such as synchronism, characters, control etc.

Receipt of printouts is rather complex, since the various reports have to be recognized and sorted at the terminals requested by the user. RT recognizes two types of reports: real printouts, destined to the concentrator terminals or to other hosts; simulated printouts of the card image, which we shall mention in the section on the SARA program.

If the process involves output destined to terminals of the computer containing SARA, the concentrator, RT orders printout by using the normal functions of the operating system; if, on the contrary, printouts destined to other hosts are involved, it activates a batch program called SARA which transmits printout into the queue of SARA.

Fig. 5 illustrates the print reception activity. The third RT activity regards analysis of completed printouts which indicate whether the output is either completed in each job has been physically concluded at the requested terminal. Fig. 7 shows control of the completed printouts.
As well as the functions described above, the RT, at this point, will send a job to the host, which includes the execution of the DECOD program, which reads the cards, prints them, and sends everything to the terminal requested by the user in step 1.

The following is a list of connections which may be affected:
- DECOD
- INVIA
- RT

The steps are as follows:
1. The first step is to send the cards to the INVIA program.
2. The INVIA program sends the cards to the DECOD program.
3. The DECOD program sends the cards to the terminal.

The final stage is to receive the printout from the terminal.
of jobs waiting to be sent to the host; those require no preliminary operator intervention, since all their functions have been automated, such as, for example, assigning of the line of communication, cataloguing of the main memory areas to be used, etc.

The modality of use of the RSETA programs which interface the user with SARA highlights the "external" aspect of this system and shows how simple it is to use.

The RSETA programs, each one of which is relative to a certain host, enable a job stream to be attached to the queue of SARA:

- print processor
- print site connected to the latter to eliminate data from a parameter card provided by the user.

Use of the system is simple, and a pre-established plan formed by five controllers of the DEC 8 system of the concentrator.

We illustrate an example of usage in fig. 1: a digit shows a job stream which is a batch terminal or conversation mode user that must execute before sending his own job to the host.

Instructions pertain to the job to be executed by the host connected via SARA whilst the part A is relative to the execution of RSETA. In particular, we shall:

- card 1 begins execution of a job on the univac line
- cards 2 and 3 fixed
- parameter card 4 indicates the name of the printer on which the user wishes to have results. Since only parameter PUE is specified, the system assumes a printer the concentrator itself.
- card 5 is fixed
- card 6 indicates the type of RSETA program to be executed. In other words, it specifies in which queue the job to be sent is to be memorized.

If, on the contrary, the user wishes to obtain printout of his job on a terminal connected to a computer different from the concentrator, the parameter card takes on the format: STRE, NOST, NCT.

where:

STRE is the name of the terminal
NOST is the name of the host processor of the network in which printout is required, e.g. CDC.
NCT is the debit code with which to execute the RSETA program by the host.

IEEE is the name of the host processor of the network in which printout is required, e.g. CDC.

5 "END"
6 "RSETA DETR1480.16M"
7 "FIN"

Fig. 9 shows an example of a job to transmit to the host.

5. THE INFORMATION SYSTEM

As well as managing the traffic of the line-up with the host, the SARA system performs the important function of gathering and filing the management data relative to jobs which are under its direct control.

A certain host has two possibilities of entering the network:

- execution, in the concentrator of RSETA, which directly recognizes the job in the queue of SARA
- execution of INVIA, by the host, which prints cards delivered at the terminal by SARA.

From a management and control point of view, the significant events which reflect on any important points in the journey of a certain job are:

- execution of INVIA: the job is placed in a queue and waits to be transmitted
- beginning of transmission
- end of transmission
- end of processing

At this point, the job has been accepted by the host and is processed.

At the end of processing, the relative printout is sent to the concentrator and recognized by one of the RT activities which sorts it.

Management data travel together with the job in the form of a control card. In this phase, the following data are taken into consideration and recorded:

- beginning of reception of printout
- end of reception of printout

The third RT activity also records the moment at which it is physically ready and the destination to which it was addressed.

In the contrary, in cases where a job is sent by a host to be executed by another host, the significant moments are:

- beginning of reception of the card deck in the form of a report, (first contact with the network).
- end of reception
- execution of CRISTA with queuing in SARA's entry queue
- beginning of transmission of job
- end of transmission of job
- beginning of reception of printout
- end of reception of printout
- execution of CRISTA which places the report, in card form in SARA's queue.
- beginning of transmission
- end of transmission

The job finally leaves the network. For each of the moments described, SARA creates an entry which is placed sequentially, i.e. in order of date, in the queue in a special historical file. The main data contained in this queue are:

- type of event (beginning of transmission, end of reception etc.)
- date and time of the entry
- name of the job in the network
Some statistical measurements on the European Informatics Network (EIN)

S. Alfarerti, S. Cenc, A. Facio, S. Palazzo, V. Soleti, G. Scalco
Istituto Elettrotecnico, Universita di Catania - CEN (Politecnico di Milano)

Abstract: This paper deals briefly with some measurements on EIN obtained by processing the statistical data collected at the Network Control Centre. The data concerns July 1979 and the results obtained allow us to characterise some aspects of the communication subnetwork performance, of the traffic on it and of the user activity.

1. INTRODUCTION

The EIN Executive Body is responsible for collecting data on user and service packet traffic in the network through the Network Control Centre (NCC). The measurements described here are taken daily (five days a week) as follows:

- In the morning (at Core Time Start (CTS)) the NCC sends to each Network Monitoring Centre (NMC) a packet to reset the statistical counters inside it; the NMC polling sequence is always the same, that is:

  1. LONDON - PARIS - MILANO - ZURICH - HELSINKI - MILANO

   At the end of the day (Core Time End (CTE)) the NMC asks each RCC for a packet containing the final counter values; here the RCC polling sequence is as above (1).

   In each RCC the counters refer separately to the activity of each different line. These lines are of two types: 1) lines RCC-NMC, which connect the NMC with another RCC; and 2) lines RCC-EIN, which connect the NMC with a Subscriber Computer (SC).

   For a RCC-EIN line the counters are as follows:

   CPM : number of user packets sent (1)
   CPMI : number of local service frames sent
   CGF : total bytes sent (including retransmissions)
   IGF : number of local service frames received
   IGF1 : number of local service frames and HPF received

   For a RCC-NMC line the counters are as follows:

   CPM : number of user packets sent
   CPMI : number of local service frames sent
   CGF : total bytes sent (including retransmissions)
   IGF : number of local service frames received
   IGF1 : number of local service frames and HPF received

   The data were elaborated in order to statistically characterize the performance of the communication subnetwork and of the traffic on the communication subnetwork, and the activity of each RCC.

2. EXPERIMENTS CARRIED OUT AND THEIR RESULTS

The behaviour of the communication subnetwork can be analysed from two different points of view: the first one refers to the subnetwork components (lines and nodes); the second one refers to the subnetwork as a single system interacting with its users, i.e. the SCs.

In order to evaluate the line performance we introduce a parameter defined as the percent ratio of the number of frames not affected by CRC error to the total number of frames received, i.e.:

\[ P_{line} = \frac{CPM \cdot CPMI \cdot CGF1}{CPM \cdot CPMI \cdot CGF1 + CPM \cdot CPMI} \times 100 \]

Such a relationship is used to characterise the RCC-NMC lines in each of the two directions and the RCC-EIN lines only in the M-M direction (2).

For each line the average and r.m.s. values of this ratio are shown in Table 1, which refers to the RCC-NMC lines, table 2 and 3 respectively show the average number of user packets per day and the average number of service packets for each subnetwork line (invariant). The satisfactory uniformity of the service traffic can be pointed out.

The subnetwork nodes will now be considered. In order to evaluate their behaviour, the following parameter is introduced for each RCC:

\[ E_{node} = \frac{CPM \cdot CPMI \cdot CGF1}{CPM \cdot CPMI \cdot CGF1 + CPM \cdot CPMI} \times 100 \]

where the sumations are extended to all the RCC-NMC and RCC-EIN lines connected to the node. This parameter gives the percent ratio of frames discarded by the node to the total number of frames correctly received. Table 4 gives the average and the r.m.s. values of this ratio for each RCC.

Fig. 1 shows the number of packet per day, in input and output on each RCC. The difference between the input and output values is due to the fact that the RCC can behave as a "sink" or (3) this is due to the lack of data on the traffic entering the SCs.

VOL.XLI - N.8 AUGUST 1979
The text describes statistical measurements on the European Informatics Network (EIN) and includes diagrams and data points related to user packets and network activity. The diagrams show graphs of user packets per day, with axes labeled as user packets per day, and various categories such as EMU, AERE, NPL, etc. The text discusses the statistical description of user activity in the network, with histograms showing the distribution of packet sizes. It also mentions the concept of source user activity and how it is characterized by the histogram of packet sizes. The text includes figures and tables with data points and statistical measures such as mean, median, and other relevant statistics. The text concludes with references to other works and measurements concerning other aspects of the EIN.
Some measurements on the EIN computer network performed at CREI
by means of the subnetcontrol module

A. Fero, G. Scella, F. Valera
ISTITUTO ELETTRONICO, UNIVERSITÀ DI CATANIA — CREI (POLITECNICO DI MILANO)

Abstract. In this paper we present and discuss some measurements on the EIN and CIDALE interconnected computer networks, performed over a five weeks period during summer 1978 at CREI (Centro Ricerca di Informatica) in Milan by means of the Subnetcontrol Module. The measured behaviour regards the subnetcontrol facilities implemented in all the nodes of the computer network. The obtained results allow us to give an estimation of some network parameters and useful information about the connection of the Italian SCs to the computer network.

1. INTRODUCTION

In this paper we slow and discuss the results of some measurements on the EIN and CIDALE computer networks, performed over a five weeks period during summer 1978. The measurements, performed at CREI by means of the Subnetcontrol Module (SCM) [1,2], have been made at the packet level and regard the addressable and non-addressable facilities implemented in all the nodes of the subnetcontrol.

Suitable commands have been given to SCM in order to produce sequences of packets directed to the addressable facilities (Virtual Subscriber Control Processors - VSCP's -); each command causes SCM to transmit a sequence of packets to a VSCP in a mode with DELIVERY CONFIRMATION (DC), VSCP DELIVERY DIAGNOSTIC (VD) or TRANCE request; the time interval between the expected departures of two subsequent packets of the sequence was selected constant.

The responses coming from the subnetcontrol to SCM were recorded on a file which was processed in order to obtain a statistical description of the subnetcontrol behaviour as seen from the SCM's implemented in the UNIVAC 1106 Host-computer in Milan.

The obtained results allow us to give an estimation of some important network parameters as the transmission time on the lines between the UNIVAC 1106 and the SC, and the time for the mapping process performed by the procedure A-Septor (about 1.5 - 2.0 s).

During the experiments which were of concern to this paper, the local terminal of SCM (called Master TT) was a TT2 at the University of Catania. From this TT2 we controlled the way in which SCM produced traffic of packets directed to the subnetcontrol. The experimental data coming from the subnetcontrol were recorded at first onto a file on the UNIVAC 1106 in Milan, then they were transferred from the UNIVAC 1106 to the CDC 6600 computer in Catania (Rome) through the link connecting those computers, in order to be processed off line by suitable programs arranged from another terminal of the University of Catania connected to the CDC 6600.

2. MEASUREMENT SYSTEM

The system available for performing the measurements on the computer network is schematized in Fig. 1. This system is constituted essentially by the Subnetcontrol Module (SCM) which is a tool studied at CREI for testing and measuring the computer network and realised by means of a piece of software running at the UNIVAC 1106 Host-computer in Milan.

SCM is an interlocutor at the packet level connected to the intermediate TT which performs the line protocol. Also TT is realized by means of a piece of software running on the UNIVAC 1106.

A procedure adapter realized by the SCELLE QA CP 360 minicomputer is put between the UNIVAC 1106 and the Network Switching Centre (NSC) because the UNIVAC 1106 has not a line procedure compatible with that of the NSC. The CF 160 minicomputer receives frames with fixed length from the UNIVAC 1106 and maps them in frames at NSC level directed to the NSC [3].

The pattern of the travelling packets between SCM and NSC is schematized in Fig. 2. Let us note that a file exists between SCM (working in demand) and TT (working in real time) in order to make them asynchronous.

The queuing structure between SCM and NSC increases the response time from the subnetcontrol. In particular the delay due to the connection SCM-NSC is constituted essentially by the following two components:

- the time for reading and writing on the files between SCM and TT (about 1.0 - 1.5 s);
- the transmission time on the lines between the UNIVAC 1106 and the NSC, and the time for the mapping process performed by the procedure A-Septor (about 1.5 - 2.0 s).

During the experiments which were of concern to this paper, the local terminal of SCM (called Master TT) was a TT2 at the University of Catania. From this TT2 we controlled the way in which SCM produced traffic of packets directed to the subnetcontrol. The experimental data coming from the subnetcontrol were recorded at first onto a file on the UNIVAC 1106 in Milan, then they were transferred from the UNIVAC 1106 to the CDC 6600 computer in Catania (Rome) through the link connecting those computers, in order to be processed off line by suitable programs arranged from another terminal of the University of Catania connected to the CDC 6600.

3. EXPERIMENTAL RESULTS

The purpose of the experiments has been to measure some network parameters dealing with the subnetcontrol facilities implemented in the nodes of the EIN and CIDALE interconnected networks (Fig. 3).

The traffic to the subnetcontrol facilities has been produced by commands emitted at the user interface of SCM which cause SCM to transmit suitable sequences of packets.

In these user commands it has been specified:
- the destination SCM to which SCM has to transmit the sequence of packets;
- the non-addressable subnetcontrol facilities requested for the packets of the sequence;
- the number of the packets of the sequence;
- the time interval between the departures of two subsequent packets of the sequence.

Then the packets were delivered by SCM to TT or received by SCM from TT, then they were recorded onto a file together with a time mark. As already said, this file has been processed off line in order to obtain the statistical description of the subnetcontrol shown in the next sections.

The time interval between the departures of two subsequent packets of a sequence has been selected

Fig. 2 - Queuing structure of the connection of SCM to EIN

Fig. 3 - Connection of SCM to EIN

VII. XII. 78 - 400 E 369 E - 543
In order to avoid congestion in the procedure adapter which produces a clock down in the transmission of packets between the INTTAC 1108 and the MILAN BSC for the reasons specified in section 2.

After some experiments the following times intervals between the departures of two subsequent packets have been selected:
- 15 s between the departures of two subsequent packets of a sequence directed to the VODAC DROP (1)
- 20 s between the departures of two subsequent packets of a sequence directed to the VODAC DROP and requesting TRACE in EIN and CIGALE
- 10 s between the departures of two subsequent packets of a sequence directed to the VODAC DROP (2) in EIN or to the VODAC DROP (3) in EIN and CIGALE.

The time interval between the emission of two subsequent commands has been selected in order to allow the reception of all the responses of a command before the emission of the next command.

3.2. TRACE

This experiment has been executed by means of commands which cause EIN to transmit sequences of packets directed to the VODAC DROP for all the EIN and CIGALE nodes with request of TRACE facility.

The nodes, which the packets pass through, send TRACE packets to EIN containing in the packet the network time in that node and the destination of the packets.

4. RESULTS

At first let us show how the subnetwork results in terms of interarrival time intervals in the experiments dealing with the TRACE and RT facilities. Fig. 1 shows the normalized distribution (1) of the interdeparture times and interarrival times intervals for the TRACE experiments. Let us note that the interdeparture distributions are impulsive like around the minimal time interval (r.m.s. of 9 s) whereas the interarrival distributions are much more spreaded (r.m.s. of 16 s) due to the internal processes of the subnetwork and to the mapping process done by the procedure adapter-dispatcher. In these experiments all the responses of the interdeparture distributions are roughly the same, but not those of the interarrival distributions because they depend on the routes of the packets in the subnetwork. The interarrival distribution for the MILAN, Pavia and Padua nodes show two peaks: the first (around 12 s) due to the TRACE experiments in EIN and the second (around 22 s) due to the TRACE experiments in EIN and CIGALE (see section 3).

Fig. 5 shows a typical normalized distribution of the interdeparture times and interarrival times intervals in the subnetwork with VODAC-BSC request. Also in this case the interarrival distribution is more spreaded, moreover it shows two peaks: the first (around 16 s) due to the interarrival of DC or RT packets.

4.1. RESPONSE TIME

The response time for both the experiments is shown in Fig. 6, Fig. 7. Let us note that the delaying normalized distribution for the TRACE responses shifts on greater values as the response come from more distant nodes (Fig. 6), in particular, in EIN the TRACE response average delay increases about 300

(1) VODAC DROP discards the packets on receipt.
(2) VODAC DROP returns a packet containing the network time in that node and the destination of the packet.
(3) RT and DC facilities are not implemented in CIGALE.

---

Fig. 3: EIN and CIGALE communication subnetworks

Fig. 4: N.1, n.2, n.3, v.1, v.2, v.3, v.4, v.5, v.6... - Normalized distributions of the interdeparture and interarrival time intervals for the TRACE experiments.

Fig. 5: n.1, n.2, n.3, v.1, v.2, v.3, v.4, v.5, v.6... - Normalized distributions of the interdeparture and interarrival time intervals for the TRACE experiments.

Fig. 6: n.1, n.2, n.3, v.1, v.2, v.3, v.4, v.5, v.6... - Normalized distributions of the interdeparture and interarrival time intervals for the TRACE experiments.

Fig. 7: n.1, n.2, n.3, v.1, v.2, v.3, v.4, v.5, v.6... - Normalized distributions of the interdeparture and interarrival time intervals for the TRACE experiments.
whereas \( p \) and \( r \) are certainly different from zero.

The way in which the experiments have been executed does not allow us to evaluate if the packet loss occurs in the subnetwork or in the connection between CSW and HEC.

4.2. LOGS AND DUPLICATION

The percentage of incomplete TRACK's due to the loss of TRACX packets is shown in tab. 3: it increases for the more distant nodes and it becomes very high in CINAE.

The duplication is a very rare event which has been observed only one in an experiment on EIN: the duplicated packet was generated by the Paris node.

The percentage of the DC, TT, and HEC responses in all the experiments excepted the last one is shown in tab. 4: the last experiment was manually performed with the subnetwork partitioned: the percentage of the DC, TT and HEC responses in this experiment is shown in tab. 4b.

Let us note that the DC packets have been always received when the relative TT packets have been received, but not the contrary.

For all the experiments excepted the last, the percentage of the incomplete transactions dealing with the HEC facility, because no answer or not complete answer has been received, is shown in tab. 5.

The sample of packets (i.e. the number of packets) is 864 for all the experiments excepted the last, and it is 250 for the last experiment. In the first case the following parameters can be well estimated [6]:

\[ p \] : probability that a packet sent to a VTRP is lost;
\[ d \] : probability that a DC is lost;
\[ f \] : probability that a packet created by a VTRP (that is a DC or HEC packet) is lost.

In the second case the following parameter can be well estimated [6]:

\[ z \] : probability that a HEC packet is lost.

Tab. 6 shows the estimated values with confidence levels of 99% and 95%. Let us note that the probabilities \( d \) and \( z \) could be also equal to zero.

5. CONCLUSIONS

The purpose of this paper was to present the results of some measurements on the subnetwork facilities of the EIN and CINAE computer interconnected networks.

The measurements were taken at EIN in Milan by means of the Subnetwork Control Module managed by the TPY at the University of Catania. The collected data can be used to characterize the network behavior as seen from the UNIVAC Host-computer in Milan; in fact the measurements were done only for the UNIVAC Host-computer in Milan.

Fig. 5 - Normalized distributions of the interdeparture (a) and interarrival (b) time intervals for a 27 experiment (Durham).
Tab. 2 - Delay (in seconds) of the DC, NT and NMD packets: average value $\mu$ and r.m.s. $\sigma$.

<table>
<thead>
<tr>
<th></th>
<th>E</th>
<th>L</th>
<th>I</th>
<th>F</th>
<th>H</th>
<th>EIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>9.89</td>
<td>10.09</td>
<td>9.77</td>
<td>11.31</td>
<td>9.75</td>
<td>9.76</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>6.27</td>
<td>5.95</td>
<td>6.30</td>
<td>6.54</td>
<td>6.02</td>
<td>6.19</td>
</tr>
<tr>
<td>$\mu$</td>
<td>11.37</td>
<td>11.32</td>
<td>11.26</td>
<td>11.31</td>
<td>11.77</td>
<td>11.33</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>6.83</td>
<td>6.50</td>
<td>6.11</td>
<td>6.72</td>
<td>5.37</td>
<td>6.59</td>
</tr>
<tr>
<td>$\mu$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4.43</td>
<td>4.43</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2.78</td>
<td>2.78</td>
</tr>
</tbody>
</table>

Fig. 7.1. a) b) c) d) e) f) Normalized distributions of the DC packet delay.

Fig. 7.2. a) b) c) d) e) f) Normalized distributions of the NT packet delay.

Fig. 7.3. Delay of the NT packets after the DC packet from EIN.

Fig. 7.4. Delay of the NMD packets from Milan.
Tab. 3 - Percentage of incomplete TRAC's

<table>
<thead>
<tr>
<th>E</th>
<th>L</th>
<th>I</th>
<th>F</th>
<th>EL</th>
<th>E1</th>
<th>E2</th>
<th>E3</th>
<th>E4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>5.9</td>
<td>0.0</td>
<td>3.9</td>
<td>26.25</td>
<td>67.14</td>
<td>67.53</td>
<td>68.57</td>
<td>95.35</td>
</tr>
</tbody>
</table>

Tab. 6 - Percentage of the MT (a), DC (b) and iDDC (c) responses for all the experiments except the last.

<table>
<thead>
<tr>
<th>E</th>
<th>L</th>
<th>I</th>
<th>F</th>
<th>M</th>
<th>E1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>96.43</td>
<td>94.13</td>
<td>92.96</td>
<td>95.46</td>
<td>93.86</td>
</tr>
<tr>
<td>(b)</td>
<td>97.62</td>
<td>98.14</td>
<td>99.99</td>
<td>96.35</td>
<td>98.65</td>
</tr>
<tr>
<td>(c)</td>
<td>7.00</td>
<td>1.03</td>
<td>5.10</td>
<td>0.00</td>
<td>-</td>
</tr>
</tbody>
</table>

Tab. 7 - Percentage of routes followed by the packets in EIR.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Roster</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zurich</td>
<td>M-E</td>
<td>100.00</td>
</tr>
<tr>
<td>London</td>
<td>M-H-L</td>
<td>99.50</td>
</tr>
<tr>
<td>M-C-P-L</td>
<td>9.50</td>
<td></td>
</tr>
<tr>
<td>Ljus</td>
<td>M-L</td>
<td>96.00</td>
</tr>
<tr>
<td>L-E-P-1</td>
<td>3.16</td>
<td></td>
</tr>
<tr>
<td>Paris</td>
<td>M-E-F</td>
<td>88.74</td>
</tr>
<tr>
<td>L-C-F</td>
<td>11.26</td>
<td></td>
</tr>
</tbody>
</table>

Tab. 9 - Measured clock rates of the network time (in nano).

<table>
<thead>
<tr>
<th>From the experiment</th>
<th>3-1292</th>
<th>3-1293</th>
<th>3-1294</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>371</td>
<td>374</td>
<td>387</td>
</tr>
<tr>
<td>0.9</td>
<td>374</td>
<td>375</td>
<td>388</td>
</tr>
<tr>
<td>0.8</td>
<td>373</td>
<td>379</td>
<td>392</td>
</tr>
<tr>
<td>0.7</td>
<td>374</td>
<td>377</td>
<td>394</td>
</tr>
</tbody>
</table>

Tab. 8 - Percentage of routes followed by the packets in CICLAD.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Roster</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>E3</td>
<td>E1-45-21</td>
<td>100.00</td>
</tr>
<tr>
<td>E4</td>
<td>E1-45-21</td>
<td>100.00</td>
</tr>
<tr>
<td>E5</td>
<td>E1-45-21</td>
<td>100.00</td>
</tr>
<tr>
<td>E6</td>
<td>E1-45-21</td>
<td>100.00</td>
</tr>
<tr>
<td>E7</td>
<td>E1-45-21</td>
<td>100.00</td>
</tr>
</tbody>
</table>

This work was partially supported by CNR (Consiglio Nazionale delle Ricerche) Document n.78/05727.

The paper was received on April 3, 1979.

REFERENCES

A. Bara, G. Le Moli, R. Sogna: Specificazioni di un Retr Network Control System for EIN, EIN/GRAF/S7/007
A. Bara: CIEI implementation of the EIN network, the Subnetwork Control Module, EIN/GRAF/S7/008
A. Gammari: Protocollo di linka/cell, EREH, EREH. Corso di aggiornamento sulle reti e calcolo, Politecnico di Milano, Nov.77, report n.19
G. Scollo: Realizzazione di misure su alcuni aspetti della rete di calcolatori EIN, Tesi di Ingegneria, Università di Catania, degree thesis 77
Contributors

Derek L. A. Barber, at present director of the Electronic Information Network Project, from October 1975 to June 1977 was chairman of BTIP, Ltd., and is editor of three books and has had several articles published recently. Before joining BTIP Mr. Barber was head of Information Services Branch - NPL, with responsibility for managing research in computer networks and information systems.

Frances Canesi was born in Rome in 1956. He obtained his degree in Electronic Engineering at the University of Pisa. He joined NCI in 1977 and was involved in the RPCHT project. In 1978 he joined ISAS International for Applied Systems Analysis in order to study problems concerning high-speed protocols. Since February 1979 Dr. Canesi is back at CNRCE, where he has the responsibility of the RPCHT upgrading, and also works in the theoretical field of high-speed protocols. Dr. Canesi is a member of ACE.

Donald W. Davies, graduated at Imperial College, joined the team which developed the first packet-switched network, and since 1951 he has been interested in packet switching and its use in computer networks. He is author and co-author of 3 books: "Computer Networks" and "Their Protocols" both published by John Wiley.

Carlo De Filippo received the Dr. degree in Electronics from the University of Rome and then got the specialization in Computer Engineering at the CNRCE of Pisa. He is currently working in the fields of information retrieval, computer networks and front-end networking with the Sperry Uninett E.D.P. consultant.

A. Dohi studied theoretical Mathematics, receiving a B.A. in 1964 from St. Joseph's College, N.Y., and an M.A. in 1966 from Boston College. From 1968-1971 she worked with a medical research group at Roosevelt Hospital, N.Y., developing an on-line recording system for the critically ill. She then worked with Centro Data Com. Forre, becoming communications software manager. Since 1975 she has been working on the ETP's multimedia system in Italy, in collaboration with the CNRCE in Rome and the Centro di Calcolo di Politecnico of Milan, in the field of communications software.

Giovanni Michel received the Dr. degree in Electronic Engineering from the University of Pisa. Since his graduation he has been at the Centro di Calcolo of the Politecnico of Milan and is currently working in the fields of computer architecture and system software.

Cesare Menini was born in Sorrento (Salerno) in 1947. He joined CNRCE in 1970 as system analyst and worked initially in the field of computational linguistics for the D.M.L. project (Dalla Mula Machine Dictionary). Since 1972 he has been working on the RPCHT project.

Andrea Mazzocchi received the Dr. degree in Electrical Engineering from the Politecnico of Milan and is currently working in the fields of computer architecture and system software.

angelo Mulas was born in Brescia (Bergamo) in 1945. He received the Dr. Ing. degree in Electrical Engineering at the University of Bologna in 1970. After graduation he joined the Faculty of Physics at the same University. In 1979, spent a year in the United States, at the IBM Scientific Center, Cambridge, Mass., where he worked on computer communications network design and techniques for distributed computing with reference to the RPCHT design. Since 1973 he has been leading the Italian part of the international STELLA (Satellite Transmission Experiment Linking Laboratories) project. Since 1974, he has been the manager of the Electrical and Electronics Division of CNRCE.

Maria Mongiello was born in Amoruso (Catanzaro) in 1948. He received the Dr. Ing. degree in Electrical Engineering from the University of Pisa in 1977. After graduating he joined the Telecommunications Division of the TTT, Albenga where he has been active in the Telecommunications Department and where he was held the position of head of the technical divisions dealing with special telecommunication and computer networks.

Mariolo Magelli received the Dr. degree in Physics from the University of Milan. Since his graduation he has been with the CNRCA. At the present he is working in the field of computer networks and computer performance evaluation.

Manuela Marchetti was born in 1953 and graduated at the University of Pisa in Computer Science in 1974. After working for a brief period at the University and a one year post graduate course, he joined CNRCE, an Institute of the Italian Research Council, in 1976. Since then his interests have been computer networks and distributed systems.

Ettore Pirozzi was born in Pisa in 1952. He obtained his degree in Computer Science at the University of Pisa in 1978. After some teaching at the University, he joined CNRCE in 1979. She participated in the RPCHT project, and now is part of the team of the STELLA (Satellite Transmission Experiment Linking Laboratories) project.

Leandre Lanseur was born in Louvain in 1948. Obtained degree in Physics at the University of Louvain in 1970 and for the whole year 1970-71 as a research fellow at the Massachusetts Institute of Technology, Cambridge, Mass. In 1971, spent a year in the United States, at the IBM Scientific Center, Cambridge, Mass., where he worked on computer communications network design and techniques for distributed computing with reference to the RPCHT design. Since 1973 he has been the Italian part of the international STELLA (Satellite Transmission Experiment Linking Laboratories) project.

VOL. XLVIII - N. 8 - AGOSTO 1979

379 E — 553
Contributors

Alberto M. Replichini, born in Roma in 1945, received the degree in Electronic Engineering from the University of Roma in 1969. From 1969 to 1971 he was assistant with the Istituto di Automatica di Roma, where he was concerned with system theory problems.

In 1972 he joined SIP - Direzione Generale, here he has been associated with the Data Transmission Department. Now he is responsible of design of dedicated packet and/or message switched networks.

Mr. Replichini is member of EUIT and CEPT groups for the packet service standardization in public data network and is also involved in EIN project, as member of TAF, and in EURONET project, as consultant of the Equipe de Projet.

P. Schicker received his Diploma in Mathematics in 1967 from the Swiss Federal Institute of Technology (ETH - Zurich). He spent one year at M.I.T., Boston, on the MULTICS project MAC, returning to the ETH-Z to head the systems software group of the computing center. His design of the terminal subsystem VENUS for the CDC 6600 earned him a Ph.D. in Mathematics in 1976. In 1973 he became project leader of a Swiss research group participating in the European Informatics Network. Since then he has been active in the area of protocol design and network architecture, coupling numerous articles with a high degree of experimentation. He has directed the ETH-network connection which today provides daily production service based on the Zurich virtual Terminal design. He is now on leave of absence from ETH to work on higher protocol designs with Bell Northern Research in Ottawa.

Marco Sommali was born in Roma in 1947. He obtained his degree in Mathematics at the University of Pisa in 1971. In 1972, he joined CNRUE as system engineer.

Since 1975 he has been working on RPCNET and, in particular, in the applications area.

Fabiio Tarini obtained his degree in Physics at the University of Pisa in 1971; after teaching for a while at the high school, he joined CNRUE, an Institute of the Italian Research Council, in 1972, as system engineer.

Since 1974, he has been manager of the Computer Network Section of CNRUE. His present interests are high level protocols and functionally distributed system design.

Richard Winsborough received a D.Phil from the University of Oxford in 1971. Since that time he has been engaged in the design of reliable multiprocessor systems, process control systems and the development of high level protocols for use on data communications networks.

Marco Zagolina received the Dr. degree in Mathematics from the University of Milano. Since his graduation he has been with the CILEA, where he works in the fields of computer networks and computer front-end.