TECHNICAL REPORT No. 30

REVIEW AND RE-ASSESSMENT OF ECNET

A PRIVATE NETWORK WITH OPEN ARCHITECTURE

by

A. Haag, F. Königshofer and P. Quoilin

May 1982

C O N T E N T S PAGE

ABSTRACT

The European Centre for Medium-range Weather Forecasts (ECMWF) has to handle on its telecommunications network (ECNET) a considerable volume of data. For this a Network Front-end Processor (NFEP) has been implemented, using manufacturer independent layered protocols, and in the present phase, medium-speed leased lines to the 17 Member States are gradually being installed and put into operation. The paper summarizes the development of the project and decisions on its architecture and the selected protocols; it attempts a review in the light of the experience gained and the present state of thinking in open inter-connection of (computer) systems.

This is the updated version of a paper presented at:

International Symposium COMNET '81

Sponsers: IFIP and UNESCO

11-15 May, 1981, Budapest

1. DESCRIPTION OF USER ENVIRONMENT

ECMWF in Reading, United Kingdom, is a joint institution of 17 European States and is governed by a Convention which came into force in November 1975.

The first priority in the Centre's work in the early years of its existence has been to implement operational medium-range weather forecasting (i.e. 4-10 days ahead) on a routine basis. This required the development of complex numerical models of the atmosphere, capable of predicting the evolution of atmospherical patterns for several days ahead. To produce results from such a model in real-time requires a very powerful number-crunching facility, and accordingly a CRAY-1 computer system was acquired, front-ended by a CDC Cyber 175, with special software for high performance job and file transfer between these mainframes.

The meteorological observational data required to define the atmospheric state for the initialization of the forecast are transmitted on the World Meteorological Organization's Global Telecommunications Network (GTS) from and to which national meteorological services take and contribute data. ECMWF receives these data from two GTS nodes on this network (the Regional Telecommunication Hubs at Bracknell and Offenbach) in continuous mode, amounting to about 6 million octets of data per day, with subsequent pre-processing and analysis.

The forecasting model executes during the night and as it proceeds, forecast data (in grid point format) tailored to Member States' requirements are produced, and transmitted to the Member States via ECMWF's own network (ECNET). The assumption is that each country gets roughly 1-2 million octets of data per night.

In addition to this, Member States should have RJE access to the Centre's large computers, and a limited interactive facility for status enquiry and operator communication should also be provided.

2. CONSIDERATIONS IN CHOOSING PROTOCOLS AND ECMWF'S NETWORK FRONT-END PROCESSOR ARCHITECTURE

When ECMWF had analyzed its expected requirements for data transmission between Reading and the national meteorological services, one of the possibilities was to use future public packet-switching services, especially considering the major night dissemination schedules with the advantage of cheap off-peak charges. However, it was clear that no reliable and timely solution could be offered, not to mention the necessity of covering the 17 Member States. Taking the amount of data into consideration, there was not even an economical benefit to be foreseen in the

immediate future. Therefore, in 1977, it was decided to construct a 'star' network, consisting of point-to-point leased lines between ECMWF and the national meteorological services, based on the X25 recommendation to stay compatible with public network developments. The X25 requirement led to our careful evaluation of all three levels of this then emerging standard and also to our wish to take part in the search for international standards on the higher levels.

On the other hand, we preferred to leave our mainframes independent of the network even as far as keeping the manufacturer's standard communications software. The support of standard system software on two large mainframes (CRAY-1 and Cyber 175) required quite substantial effort, with the CRAY-Cyber link project also under development, so on balance we considered it an easier task to limit our remote communications development to a Network Front-End Processor (NFEP) with standard CDC/INTERCOM protocols towards the Cyber machine.

With the different set of procedures towards the Member States, i.e. the "ECNET", and towards the Front-End (FE, i.e. the Cyber in our configuration), a further simplification led to the concept of spooling whole files over disks attached to the NFEP. This decision was made possible by the nature of most of our applications, such as RJE or data dissemination. The latter was expected to run fast, or even with priority, on a schedule, but indeed not exactly in "real-time". The concept of disk spooling was supported by using the possible high throughput rate on the NFEP-FE link, a much higher throughput than the combined throughput on all communication lines, thereby reducing file disposal times from the FE with corresponding reduction of FE-system overhead. This also permitted the stand-alone running of the NFEP in case of scheduled or unscheduled FE breaks. Moreover it provides the basis for an independent tuning of the FE-NFEP batch data stream throughput. The concept of independent file spooling by the NFEP lends itself to the sharing of disks by NFEP and FE, which would eliminate the need for FE-NFEP file transfers and the corresponding delays. Such a feature was, however, not available to us and therefore did not play a major role in the design.

From the network point of view the FE-independence of protocols liberated us from some restrictions of the FE software. We could, for instance, define a file checkpoint and restart procedure: we could enrich the file types and define transparency or non-transparency (i.e. ASCII) of data as per logical record. Ultimately, however, the FE interface imposed its limits on the variety of applications and possible throughput.

The communication architecture chosen by ECMWF to handle ECNET was a hierarchy of

manufacturer independent protocols: LAP B of X25, an end-to-end protocol developed from INWG.96 [1], an own file transfer protocol including RJE, and a limited interactive facility. As stated above, the NFEP was to be coupled to the Cyber 175. The software in the NFEP had to be structured like a watershed and our contractor, SIA Ganymede, London, had to develop software for handling CDC INTERCOM protocols as well as ECMWF's protocols (towards ECNET). These protocols and the NFEP architecture are described in more detail in another paper [2].

Each Member State was free to choose (but also responsible for) its own termination of the line to the Centre. To give an example: two possibilities, i.e. a smaller stand-alone terminal and a direct mainframe termination of the line, were examined. It should also be mentioned that on the end-user level, graphics played a most important role in using weather forecast data and system selection and implementation had to take this into account.

The present state is that the NFEP with all its software has passed final acceptance. Operational forecasts have been produced and disseminated since 1st August 1979, and on a daily basis since 1st August 1980. Table 1 lists implementations in our Member States (to be) connected with medium-speed lines (2400 or 4800 bps). It shows the 'open' character of the ECNET architecture, where individual responsibility for the connection of network hosts is retained, a feature, which was also characteristic in the development of the much earlier GTS network.

Meteorological Service of	Hardware Hardware	Software developed by	Status (31.3.81)
United Kingdom	Ferranti Argus 700E/G	U.K.Meteorolog. Service	Operational
F.R. Germany Sweden	RC3600	SIA Ganymede	Operational
France, Austria Finland	CDC Cyber 17x (NOS/NAM)	French Met. Service	under test
Ireland	PDP11/DEC20	Trinity Col. Dublin	Operational
Denmark	RC8000	Danish Met. Service	under impl.
Greece, Portugal	CDC18/20 mini	CDC La Jolla	under impl.
Netherlands	Burroughs B800/B6800	Dutch Met. Service	under impl.
Belgium	Philips DS714 /81 mini	Philips	under study
Italy, Spain, Yugoslavia, Switzerland Spain	not yet committed		

TABLE 1. Hardware and software developments to connect Met. Services to ECNET.

In order to bridge the time until development and/or acquisition of medium-speed line terminating hardware and software Member States could use interim telegraphic circuits, practically without transmission protocol (and error control) at all, for the reception of forecast data only.

3. A REVIEW AND RE-EVALUATION OF ECNET'S PROTOCOLS

The Physical Layer

X21bis instead of X21 had to be chosen as the only practical solution at the time.

The Data Link Layer

LAP B was selected but a procedure for its initialisation in a point-to-point (DTE to DTE) mode had to be defined, following a proposal by our hardware contractor (A/S Regnecentralen). Various, now well known, omissions of the LAP B procedure were rectified, in line with the corrections applied in the various emerging public networks.

During implementation of the NFEP and in the Member States, a few common misconceptions (e.g. internal frame and octet presentation) could be identified. The official LAP B document by the CCITT is still not 100% accurate, but certainly allows to achieve compatible implementations. In our rewrite of this document we unified DCE and DTE treatment, aiming at symmetry where applicable. Nevertheless, this protocol seemed for everybody in ECNWT the most difficult one of our three layers to implement.

The Network Layer

We would have selected level 3 of X25 but this has not been implemented due to our initial set-up with only point-to-point connections. The usage of level 3/X25 to support end-to-end flow control and message fragmentation was intensively discussed by us during the early design stages, but was not adopted for various reasons, some of which are now quite common, e.g. the missing end-to-end significance (if used on a PPSN) and the need to send "messages" as consecutive streams of packets which undermines the usability of X25 circuits to multiplex several higher level "liaisons". ECNET is anticipating a possible level S/X25 usage in all its software designs, but then X25 would be confined to its network (DTE-DCE) flow control and its host addressing functions. In this way, permanent virtual circuits or virtual calls might be used quite liberally.

The End-to-end Layer

We simplified the INWF.96 proposal 1 by eliminating lettergrams and reducing the number of commands. Flow control, initialisation and termination were much refined. An application service field was later introduced into the in-

itialisation command.

Our practical implementations have shown that our end-to-end protocol keeps integrity and sequence of data very well. This is, however, assisted by a strong underlying data link level, which on our point-to-point connections already has a high level of end-to-end control. Our end-to-end protocol has even demonstrated a useful degree of redundancy by overcoming many situations correctly where working was not strictly according to specifications.

With regard to flow control on behalf of the application level, implementations in our Member States were not usually consistent with the intention, e.g. flow control is exercised without regard to the application level, or application level knowledge is somehow existent in the end-to-end layer to make proper anticipation of flow.

Some implementations showed too much dependence of applications on the (often only temporary) malfunctioning of the data link layer. The end-to-end layer was in general not implemented as an effective shield, and faults on the link level were allowed to "crash" the applications (i.e. the file transfers) too early. In the whole ECNET protocol hierarchy, the end-to-end layer has the responsibility for data integrity and therefore malfunctions on its own level must lead to serious repercussions on the higher level. But for the same reason it could be made very robust indeed against data link level problems.

Finally, in INWG.96 as well as in our end-to-end protocol we never found the fragmentation mechanism entirely satisfactory as it demonstrates the dependence on the requirements of the underlaying levels, but all studied alternatives led into other problematic side-effects.

The File Transfer and Interactive Layer

We did our own file transfer protocol development based on emitterforced file transferring. It was kept very simple to permit efficient flow of file data. From the various choices for the data transfer unit we chose the buffer ("letter") and not the logical record, a concept we however retained to support a minimum of file substructuring. Implementation and operation showed that with the provisions we have on the end-to-end level no separate flow control on this level was needed. Even restart after recovery could be solved without the need for checkpoint acknowledgements.

However, this level revealed in practice many problems connected with the proper selection and the timely start of file transfers. These problems relate to the desire of the emitter to get rid of his files as compared to the possible inability or "unwillingness" of the receiver to accept them. For instance, the receiver may be switched off or might have trouble with a required peripheral.

We have now introduced an application service field carried by the initialisation command of the end-to-end liaison to solve most of these problems. Via this field the called application (in this case the file receiver) can give a minimum of status-information already at end-to-end liaison set-up time and thereby guide the initiator on a proper path to select a suitable file (or suspend action) before an actual transfer is attempted. The recent introduction of multi-streaming of files (parallel transfer of more than one file per direction between Member States and ECMWF) also aimed at the aforementioned problem. Multi-streaming however has also some negative side-effects, i.e. increased resource-consumption (buffer memory) and overheads, and does in general not increase line throughput. This proviso is especially true for implementations with disk-spooling of file transfers. Implementation of multi-streaming is therefore not mandatory in ECNET hosts.

4. ECNET PROTOCOLS COMPARED TO ISO OSI THINKING

Being so much influenced by INWG.96, our end-to-end protocol combines session control and transport control functions. The whole fragmentation process of INWG.96 could be compared to the isolated transport function of the OSI model where the network independent "letter" (transport service data unit) is adapted to the network dependent "fragment" (transport protocol data unit). However, we have excluded flow control from the fragmentation level.

On our leased point-to-point lines the data link level (LAP B of X25) serves transport end-to-end functions and it could be questioned whether the extra flow control provided is needed or useful at all. But as there is the other aspect of data link flow control, i.e. limiting the actual speed of transmission on the communication links in case of a too slow communication processor or host overload, this flow control can probably not be relinquished. Also, at least in some of our Member State's implementations, the communication processor is not the "end" in the sense of the end-to-end protocol, but a mere transit stage from data link to data link.

On the higher levels, it would have been hard to justify a general purpose

session layer checkpoint/restart mechanism and/or extensive presentation controls, due to the limited range of applications foreseen, as well as the limited facilities supported by INTERCOM.

In summary, the OSI model was of influence to us when we were confronted with a practical problem during or after implementation and needed to identify the level (layer) on which to tackle or solve it. But even with our quite simple selection of layered protocols, it was already difficult to pack the major functions in a compatible way into smaller stand-alone terminals. Performance or reliability targets have, however, always been fulfilled.

5. REVIEW OF INTERNAL INTERFACES BETWEEN PROTOCOL LAYERS (NFEP)

Based on a multi task operating system, all protocol handlers had been implemented as single processes, communicating with each other via message-answer sequences. The interface between the processes was designed to be as simple and as secure as possible to protect the whole system against malfunctioning of a single process.

To achieve this, it was decided to have different "streams" of information transfer between the layers, one control stream and several data streams (one for each logical link). It soon became clear that on the data streams a minimum set of primitives would be sufficient for our purposes. It was then decided to define primitives which are independent of the protocol layer (i.e. do not reflect the data structure of a specific layer), and hence it was possible to use similar interfaces between the different layers.

A very strict sequence of primitive exchange was defined, allowing processes only to exchange data after a request from the receiving process. This was to ensure a certain amount of flow control between the layers and to avoid at the same time long message queues waiting for a 'slow' process.

Due to the similarity of interfaces between different layers it was possible to implement a set of co-routines (called 'message handler') which handle all exchanges of primitives between the layers. This approach proved to be very valuable as it firstly avoided implementation of equal code in every process and secondly it provided a very good means for implementation of a 'message trace' facility, which was then heavily used during the debugging phase (see para.6).

The impelmentation of this design has proved that this was the correct approach

towards the problem, as the simplicity helped to prevent logic errors and the strict directives on the sequence of primitive exchange assisted the debugging process.

6. TESTING FACILITIES FOR NFEP SOFTWARE AND EXPERIENCE WITH THE SET-UP OF NEW LINKS

Provisional and Final Acceptance approach (NFEP)

For the Provisional Acceptance criteria were defined which had to be met by the delivered hardware and software. To demonstrate that the criteria are met, trials were defined to test the complete system in all respects. The trials were selected to reflect the envisaged operational usage of the system, hence most software tests had been designed to demonstrate the correct working of the system under normal circumstances. Nevertheless, tests had to be included which checked the system right to its limits, especially with regard to the maximum throughput and the safety against malfunctions or 'anarchic' behaviour of a remote terminal. One important criterion therefore was that under no circumstances a misbehaving remote system could bring the NFEP down.

Overall the Provisional Acceptance was designed to be a rather short (24 hours) and single event, whereas the Final Acceptance - lasting for a period of 9 months should prove that the system is capable of operating reliably and according to specifications. The main criterion of the Final Acceptance therefore was the fulfilling of a certain grade of reliability expressed in the figure of 'Overall Availability', defining the percentage of time the system had to be available for operational use.

Hardware test facilities

As the NFEP hardware was well supported by the manufacturer, the main concern lay on the provision of test facilities for the network itself. First a line was drawn between the analogue side (modem to network) and digital side (modem cable to NFEP). A patch and jack field was designed for both sides allowing easy access of test equipment to the line, the modem and the port as well as easy switching possibilities of ports, lines and modems. On the analogue side a selection of test equipment ranging from a simple oscilloscope to advanced line distortion metering devices were installed to enable a fast location of faults. On the digital side, the main test equipment installed was a data analyser, capable of interpreting the lower levels of our protocols (X25).

Software test facilities and set-up of new links

The first stage in setting up a new link is the checking of the line itself. This usually is done by switching the remote end to loop back and monitoring the quality of a random bit pattern sent across the line. Once it has been established that the line is of acceptable quality, the first test transmissions can start.

During this phase a great deal of support has to be given to a Member State as usually the remote end has little or no means to monitor what is happening on their line. It is especially important to have a good means of communication between the two ends for quick reports on test results, problem reports etc. It was therefore decided to use the 'Secondary Channel' of the V24 standard as communication link. All it needs is a TTY operating at 50 baud at both ends to provide for this simple means of human to human communication and a hard-copy record of this dialogue.

The phase of the first actual data transfer across the line is heavily assisted by the use of the data analyser and software tests built in the NFEP system. The data analyser is mainly used to detect problems occurring with the X25 protocol. Each test session is recorded and in case of a problem the recording can be analysed and the error can be detected quickly.

Problems with the higher protocol levels usually are detected by the NFEP software itself with console messages generated accordingly. Nevertheless, the data analyser recording often comes in handy to prove 'whose fault it was'.

If the error still cannot be established with the facilities mentioned so far, a total 'message trace' can be switched on in the NFEP, which will record all internal message and data flows between the different protocol layers. This trace is then processed by off-line utilities to give an easily readable report.

It should be mentioned that for a test session it so far was not necessary to arrange special data files for the transmissions. It was always possible to use ordinary remote job files or some of the currently available forecast files. The experience up to now is that it usually needs 4 to 5 test sessions of a few hours duration to get a remote terminal working reliably enough to send and receive complete files. Special tests have of course to be arranged to test specific error or timeout situations.

Overall, it can be stated that the data analyser grew to the most important testing tool during the set up of new links. It is capable of showing exactly what was going on on a link. Never can there be any doubt about whether a data block really was sent or not. Due to the fact that the data analyser is capable of displaying X.25 data in a very easily readable form it is mainly used to trace problems on these levels. Nevertheless, as it is the only testing tool which has no influence whatsoever on the data exchange ('message trace' for example slows the NFEP down considerably) it is of great help in tracing problems in all layers, especially if a problem only occurs in real time situations.

7. REVIEW OF PERFORMANCE AND SYSTEM TUNING (NFEP)

As the NFEP system from the start was designed to achieve a very high data throughput, not many problems were experienced in respect of performance. Nevertheless, when the system first came into operation the performance was slightly below the expected values. Careful investigation revealed that most of the delays were caused by lack of buffers and unnecessary protocol overhead on the end-to-end and file transfer level.

The buffer shortage was regarded as a serious problem as it would have meant either to redesign the system or to try and 'squeeze' the software as much as possible. There was a third possibility, i.e. to increase the memory on the NFEP, and as this was quite a cheap and clean solution it was adopted.

In addition to this a thorough reconsideration of the acknowledgement procedure of the end-to-end and file transfer levels led to improved protocol handling which brought the performance to the expected values.

Currently, in a situation of moderate usage (3-4 links with data transfer) the data throughput lies in the range of 70% of the line speed with an idle line in send direction of just above 10% of the total transmission time. Therefore, the total protocol overhead of all layers together amounts to not more than 20% of all characters (including flags) transferred, which is regarded as very good.

Conclusively, it can be said that, in spite of a few 'teething' problems, due to the careful design the anticipated performance could be achieved without redesign or rewriting of software.

8. REVIEW OF OPERATIONAL CHARACTERISTICS

Operational Control

With the increase of traffic load at the NFEP due to new Member State connec-

tions, malfunctions caused by the network, the NFEP or the CYBER/NFEP link had to be identified quickly. An 'operational watch' program was therefore developed displaying an up-to-date picture of NFEP operational aspects such as the general status of the NFEP, the state of the CYBER/NFEP connection and for each Member State link, the line status, the line quality (under implementation), the number of files remaining in each specific queue, the name of the file(s) in transmission, the file transfer protocol status and the status of the interactive liaison.

Little has been implemented yet in the area of gathering and displaying line and traffic statistics but this will receive attention in future.

Reliability

The software proved to be very stable. Although system crashes due to software errors did inevitably occur as our system moved towards its full operational environment, those crashes were handled by an automatic recovery procedure which promptly restarted the system and provided the analyst with a dump catalogued on disk.

There are always transient phases of unstable operation of lines or equipment at ECMWF, as well as in the Member States, and restart procedures (using the "checkpoints" of the file transfer protocol) proved to be very important in preventing such phases of instability to impair the overall operation.

On the hardware side we had very few troubles and most of the NFEP is duplicated anyway in the form of a secondary computer used for development of the software and to assist the link set-up phase of new medium-speed lines. As HDLC controllers are quite a new hardware (and firmware) item we experience occasional problems and manufacturer improvements applied to this particular hardware.

Our results with lines and modems are quite positive. We found the lower quality M1040 leased circuits as adequate for our operation at 2400 or 4800 bps and also the modems of sometimes different manufacturers co-operate without problems.

9. SUMMARY AND OUTLOOK

The architecture and the protocols of ECNET have been found adequate so far for the data dissemination and RJE requirements of ECMWF. They, and in particular the NFEP concept, have created a stable environment for the further evolutionary development of ECNET, i.e. the addition of a few useful features and the gradual

introduction of medium-speed lines to ECMWF's Member States. In this stable environment it is also quite easy to think through some further additions to the concept in an orderly way, e.g. the possibility of adding a network level (X25 level 3) for some or all of the lines and the increase in network statistics and network control facilities. As now several projects to link Member States to ECNET have come into production or are close to this state, future enhancements will require a considerable effort in consultation and co-operation.

The NFEP concept and the ECNET protocols have, however, also some drawbacks which are worth recalling. The necessity to keep the INTERCOM software in the FE (the Cyber) and the total decoupling of the Network from the Cyber results in a lack of feedback between Cyber/CRAY and the Member States as far as the exact location of a file transfer from ECMWF to a Member State is concerned. An end-to-end control from ECMWF-mainframe to Member State application is not practical to implement but would be quite welcome. The file transfer protocol lacks a file call-up facility which would cost a lot of effort to develop. The end-to-end protocol serves very well in ECNET's leased line environment and would probably do well in packet-based network services but it is unclear how it could serve above high-speed transmission systems e.g. satellite or local broadband communication.

A next step at ECMWF could be to consider the local interconnection of mainframes and special processors as that might become a pressing problem in the years to come. The ECNET architecture is quite a good starting point for this in its good flow control principles and its manufacturer independence. Apart from the drawbacks mentioned above the questions of mainframe software interconnection and user interfaces to such a local network (and to the remote network) and questions of performance would require special attention.

ACKNOWLEDGEMENTS

Many experiences and conclusions described in this paper are strongly related to the work of the implementors of the NFEP software and to other early implementors of ECNET software in Member States. John O'Leary of SIA Ganymede, project leader of the NFEP project, delivered an excellent professional product and it is hard to adequately acknowledge the effort he personally had put into the success of the project. We also happily acknowledge the special contributions by Philip Rakity, deputy project leader, and by James Aitken, project leader of the RC3600 terminal software project for the Meteorological Services in Germany, Sweden and Denmark, and by John Smith who was responsible for the implementation at the

United Kingdom Meteorological Office. Discussions with them gave ECNET its shape.

REFERENCES

1 Cerf, V., Mackenzie, A., Scantlebury, R. and Zimmermann, H. 'Proposal for an Internetwork End-to-End Protocol', International Network Working Group (INWG) General Note 96, May 1975, with later revisions.

2 Königshofer, F., and Rakity, P. 'Network for Meteorological Applications', Symposium NETWORKS 80, 4-6 Feb. 1980, Bombay, India, 97-108.

GLOSSARY

Application Service Field (ASF)	a text field in the initialisation command of the transport layer; used by an application to inform the remote user about the availability of services;
Batch data stream	the data stream on which job input or job output are transmitted;
Baud	unit of the signalling rate on a communication line; depending on the transmission technique used, it does not necessarily equal bits per second.
CCITT	Abreviation for "Comité Consultatif International Télégraphique et Téléphonique"
Communication protocol	a set of rules by which two interlocutors communicate
Co-routine	a routine which, when activated, is re-entered at the point where it last relinquished control: supports quasi-simultaneous execution of program modules.
Data analyser	a piece of equipment capable of analysing the data traffic on a line without interference
Data dissemination	the transmission of meteorological products to a Member State
Data link layer	the lowest non-physical protocol layer; responsible for error free transmission of bit streams; usually referred to as level 2
DCE	Data Circuit-terminating Equipment; the network side of the interface to a network or line; usually the modem, but might include higher levels as well

DTE	Data Terminal Equipment; the user side of the interface to a network or line
ECNET	ECMWF's telecommunications network
End-to-end protocol	the protocol responsible for an error free communication between two end users (applications) over a network; usually referred to as level 4
FE	front end; the Cyber 175 computer at ECMWF
File transfer protocol	the application layer protocol responsible for error free transmission of files
Flag	a data link layer protocol element; a flag marks the beginning and the end of a transmitted bit stream (frame)
Flow control	the procedure by which an interlocutor regulates the data reception rate according to its needs
Fragment	the data transmission unit of the end-to-end protocol, i.e. a data unit reflecting the needs of the underlying transmission layer
Fragmentation	the process of splitting letters into fragments
Front end	as opposed to back end; a computer connected as a pre/post processor to another (usually more powerful) computer
HDLC	High Level Data Link Control; a data link layer protocol standardised by ISO
Interactive facility	an application allowing user-to-user and user-to-computer dialogue
Interlocutor	the module (hardware or software) interacting with a corresponding module in the other system (e.g. remote system) on the same layer

INWG	International Network Working Group; the Working Group 6.1 of IFIP
INWG.96	INWG's General Note No.96, the proposal for an end-to-end protocol
ISO	acronym for International Organisation for Standardisation
LAP B	Link Access Protocol - Balanced mode; a data link layer protocol standardised by CCITT and compatible with the balanced class of procedures of HDLC; the level 2 of the X.25 standard
Letter	the data transport unit of the end-to-end protocol, i.e. the data unit on which the end-to-end protocol performs flow control and error control; of almost abitrary length and and comparable to the known terms buffer, record or message.
Lettergram	a mode of operation of the end-to-end protocol where data can travel from end-to-end without having had a logical path (liaison) set up; not implemented in ECNET
Liaison	a logical link between two end-to-end interlocutors
Medium speed leased line	in ECNET a 4 wire circuit rented from the PTT; transmission speeds of 2400 or 4800 bps.
Modem	abbreviation for modulator - demodulator; a piece of equipment transforming digital signals into analogue signals, as required on medium speed lines, and vice versa
Multistreaming	the process of transmitting several files simultaneously over one line
M1040	CCITT standard for normal quality medium speed leased lines

NFEP	Network-Front-End-Processor, the Regnecentralen 8000 system at ECMWF
OSI	abbreviation for Open Systems Interconnection, i.e. a manufacturer in dependent architecture of systems interconnection as worked out by ISO
Permanent virtual circuit	one kind of logical connection in the X.25 level3 standard, not requiring call set-up and clearance (analogue to a leased line)
Point-to-point connection	a connection between two DTEs without a network in between
PPSN	abbreviation for Public Packet Switching Network
Primitive	a command or response exchanged between processes
Protocol handler	a process handling the communication protocol of a layer
Protocol overhead	the overhead caused by the transmission of protocol handler etc.
Secondary channel	a low speed, asynchroneous, independent second channel provided by the V.24 standard, filtered out by the modem on a medium speed line
Session layer	layer 5 of the ISO OSI reference model; responsible for the management of application to application communications
TTY	abbreviation for Teletype, an asynchroneous input/output device
Virtual call	one kind of logical connection in the X.25 level 3 standard; analogue to dialled telephone lines
V.24	the lowest physical layer protocol; defines the interface for bits-passing between modem and computer

X.21, X.21 bis	physical layer of the X.25 standard; X.21 bis is compatible with the old V.24 standard: X.21 is the standard for (future) digital telecommunications interfacing
X.25	CCITT standard for publick packet switched networks; 3 layers - physical, link and packet level.