IMPLEMENTING THE ISO-OSI REFERENCE MODEL

by R. Popescu-Zeletin

Bereich Datenverarbeitung und Elektronik
Hahn-Meitner-Institut für Kernforschung Berlin GmbH, Berlin, Germany

Abstract

The ISO-OSI reference model was the design framework for the implementation of a heterogeneous network in a local environment. The HMINET-2 is a network based on standards in data communication. The paper focus on the OSI model feasability and outlines the encountered problems and the adopted solutions in the network design and implementation.

1. Introduction

In the last five years tremendous research and standardization efforts can be witnessed in the data communication field in order to support the requirements of standard communication features in heterogeneous environments.

The fast technological developments in the field of data communication have been materialized in a variety of products which claim to solve the user's requirements for networking. On the other hand, the users have precipitated this invasion by calling for increased connectivity, higher reliability, lower costs in networking and support for interconnecting hetergeneous equipment. The result of this situation is that the user has a difficult task in developing strategies for its environment in order to get unscathed through the product jungle of the market[6].

The research and standardization efforts have been materialized in the ISO-Open System Interconnection Reference Model.

The layered model has two main objectives:

- to provide a logical framework to bring order into a very complex domain,
- to separate functions characterizing a certain layer in order to provide independence between standards and corresponding products at different layers.

The reference character of the ISO-OSI model has and will have a dominant role in the future of standards and products in data communication since it imposes a rigorous discipline for the development of products and systems to achieve compatibility. Many organizations (CCITT, IEEE) and products suppliers (DIX etc.) have adopted the model and are referencing their systems and products to the model.

The paper outlines the experience and problems coming up from a pilot project in networking which has been sponsored by the German Ministry of Research and Technology. The scope of the project was the development of a heterogeneous network based on ISO-OSI reference model and the related standardization output.

First we outline the main characteristics of the hardware and software characteristics and the network embedment in the national and international public networks.

Chapter 3 focusses on some general conceptual problems of the ISo-OSI model encountered in the design of the network. Then we discuss the practical and conceptual problems in each layer and we describe the adopted solutions.

Some main points of concern and general conclusions are summarized in the last chapter of this paper.

We assume that the reader of this paper is familiar with the concepts and the terminology used in the ISO/TC97/SC16 documents.

2. HMINET2 - a network based on standards

HMI is a nuclear research institute in West-Berlin. The requirements for networking and distributed processing have been identified at the beginning of 70's. The result was the development of a local area network (HMINET-1) which was based on a design closely influenced by the local environment at that time. The network was a result of a cooperation between HMI and SIEMENS AG. and was installed at different places in the world[3]. The experience gained in the operation of HMINET-1 has influenced the decision to start the design of a new network called HMINET-2. The main aspects which have influenced this decision are:

- New computer types in the HMINET-1 environment impose a large implementation effort.
- New operating-system versions require changes in the network software.
- Requirement for on-line communication with other similar institutions in Germany or elsewhere.
- The requirement of a global concept which covers network and real-time applications in order to minimize the implementation effort.

These considerations and the new impacts and know-how in network technology have determined the basic aims of the HMINET-2 design:

1) - the manufacturer independence by adopting accepted standards,
2) - design of an open-ended network architecture,
3) - high interconnectivity and flexibility by allowing computer to participate at different network technologies using the same standard application
4) - compatibility with the national public network to allow communication to the outside world.

The developments in the standardization organizations, the evolution of the ISO-OSI reference model and its implication in the data communication market, the CCITT recommendations and their impact on the development of European networks are some of the facts which have influenced the decision to develop a network based on standards.

The computer types in HMI and the corresponding operating systems are depicted in the following table:

Host type	No. of Hosts	O.S.	Supplier
S7880	1	BS3000 (IBM compat.)	SIEMENS
VAX	2	VMS	DEC
PDP11 40-70	25	RSX 11M	DEC
R-30	2	ORG	SIEMENS

Outside HMI, in the Berlin area and in other institutions of interest for HMI users, a large variety of computer types are directly connected to the German public network (DATEX-P). Some of them are: IBM, CD, UNIVAC, S7700 etc.

The high degree of heterogenity explains also why the design had to rely on a reference model (ISO/OSI) and the corresponding standards.

The network topology and its embedment in the national and international public networks is outlined in fig. 1.

The adopted communication architecture and the related standards are depicted in fig. 2.

3. Conceptual problems

There are a number of issues which are not clear in the ISO-OSI reference model specification[4] for a network designer.

Some of them are:

1. Layer independence.
The adopted layering technique is characterized by two major concepts:
-The layer service which is the set of capabilities offered at the boundary of a layer to a user in the next higher layer. Note that the service is an abstraction by which the capabilities offered by a layer (in using all lower layers) are specified and that the service definition is independent of any particular implementation.
-The protocol which defines the rules of interactions between entities which are situated in the same layer but pertain to different systems.
It is obvious that by the definition of the service offered by a layer and the definition of the service offered by the lower layer the functionality of a layer is completely specified. Note that different protocols may carry out this functionality.
Although these architectural concepts aim to layer indepedence the services defined in the related standards for each layer do not provide completely this goal. This is mainly due to the fact that the services are defined as a three party communication. The three communicating entities are the two users of the service and the layer below as service provider.
The implication of this fact is that the specifications of the services provided by each layer define the capabilities offered by the layer <u>and</u> the behaviour of the two users of the service in the layer <u>above</u>.

A neutral service definition without the involvement of the behaviour of the entities in the layer above is probably the only way to achieve the aimed layer independence.

2. Multicast communication
The ISO-OSI model provides provisions for point-to-point communication only. A large class of applications requires multicast communication regimes (distributed databases, mailing and teleconferencing systems etc.). At the present time the user of the model has to solve the communication aspects for multicast applications outside the model.
Since one of the aims of the model is to relieve the users of the model of communication oriented aspects, for this class of applications the model failed its aim.

3. Layer entities
Another practical and architectural problem is the fact that in the OSI Reference Model the existence of entities in the next higher layer is assumed at connection set up time (e.g. Session Entities for the Transport Layer in the establishment phase). This assumption is not true for most implementations if the implementator cares for implementation effiency and so does not implement a multitude of dummy processes which have to wait to be activated. That means that before a CONNECT indication may occur the creation of an entity in the Session Layer has to be performed. It is not clear from the service specification which entity in which layer has to enter the termination phase if the entity cannot be created or when deadlock situations occur.

4. Dynamic change of quality of service
The present documents specify the negotiation of the service quality only at connection set up time. There are many cases where a dynamic change of the quality of service is required. In the existing documents a quality of service may be changed only by involving the termination of the connection and then by re-establishing a new connection with the new service quality. This schema is too rigid and too costly.
A practical problem in the network design is also the criteria choice on how different quality of services at different layers can be maped.

3.1 Layer 1 to 3

As already mentioned the X25 LAP B recommendation was adopted and provided by the host manufacturer for the first three layers.

The practice has shown that even if a standard is available the manufacturer are reluctant in providing it. The reluctance is motivated by the options available in a standard and also in the ambiguities contained in it. These ambiguities appear through the informal description technique of the standard. Although the informal description eases the readability of standard, a formal methodology accompaning the informal one will minimize false interpretations and implementation effort. Since the first three layers are provided by different manufacturers in the HMINET-2 a lot of effort was put in testing, certification and alleviation of incompatibilities of the products supplied by the manufacturers. This task was eased by the existence of the requirement that all X.25 products have to be fully compatible with the adopted version of the X.25 in the German public network (DATEX-P). We learned that a certification authority and appropriate testing mechanisms and procedures are vital in the development of the network.

The certification task has accompanied the whole project and even now for new communication links to hosts connected to other national public networks differences appear between national X25 implementations (e.g. facility field).

A major practical and architectural problem is the visibility of the X.25 services. The question is "where the upper layers access the X.25 layer 3 services, in host or in the front-end?" (fig. 3)

Usually between the host operating system and front-end a manufacturer specific protocol is implemented (fig.3a). Adopting the ISO-OSI model in the network design means that the upper layers have to use the services and the characteristics of the layers below. The second solution(fig3b) has the advantage that all layer 3 characteristics: flow control, error detection and recovery, addressing are available in host for the upper layers. Since the standards in transport layer are based on backpressure, error reporting etc. the second solution eases the implementation and design of the network.

Disadvantage of this solution is the fact that it is contradictory with the front-end concept of relieving the main frame from the communication aspects.

We choose the second solution since standard products in the upper layers were not available from the manufacturers at that time and have had to be designed and implemented in HMI (fig. 4).

3.2 Layer 4

The scope of the layer 4, also called transport layer is to ensure an end-to-end communication between processes situated in different hosts. The present available ISO DP 8073 identifies 5 classes of protocols and mainly mirrors the compromize between ISO and CCITT outputs.

If class 0 and 1 are supposed to support usual PTT applications (teletex etc.) class 2 to 4 are characterizing the user concerns of cost effectiveness and enlarged interconnectivitiy with networks based on a connectionless type of service. The main practical problems encountered in the network design are:

a) Choice of the class and options to be adopted. This decision is influenced by the environmental characteristics and the scope of the network. The multitude of choices and the fact that a manufacturer decision towards a certain class and set of options is inpredictable at the present time the transport layer design is a difficult task.

b) Cost effectiveness. If one of the scopes of the layer 4 is to be cost efficient then it has to provide additionally a service which will allow the User of the Transport Service to interrupt the Network Connection (for which he has to pay) without interrupting the life of the Transport Connection. That means that the Transport entities will exist even if the corresponding Network Connection is discarded. Such a situation may occur when an application transmits data from time to time and does not want to pay for maintaining the Network Connection for the time when no communication occurs.

c) Authorization. If the Transport Layer has an end-to-end significance it has to provide an authorisation mechanism for the access in the end system from remote. In the present specification no protection mechanisms (e.g. user identification, passwords etc.) are mentioned and that implies that resources may be used by everybody everywhere. An intended or unintended user misbehaviour may use all resources of the remote system. In the current specifications this misbehaviour cannot be detected[5].

Some of the above items are identified in the DP8073 but considered for further study. At least the accounting and TC user identification seems to be vital for each operational network. The decision to implement a transport layer which offers the functionality of class 1 protocol is motivated by the HMINET and BERNET environment and scope.

The time scale of the project development did not allow to wait for a standard of the transport layer. We have adopted the message link protocol (MLP) a German research environment standard which is equivalent with the ISO class 1.

The accounting problems introduced by multiplexing of transport connections over a X.25 network connection, and the fact that HMINET is a compus network (no transmission costs are charged for local communication) are some other arguments for choosing a protocol functionally equivalent with class 1.

Again the formal description of the protocol was an important aid towards compatible implementation in the different hosts[1].

The emergence of the LAN technology in data communication have influenced the decision to reconsider the implemented layer 4 protocol. Since the tendence is a decentralization of the network in closed LANs for specific applications and the HMINET-2 use as a back-bone network (fig. 1), connection and connectionless services are required at the transport layer boundary. The most suitable class in DP8073 specification is class 4.

The advantage of a rigorous layered architecture for the network implementation is obvious since the changes introduced do not influence the layers above and below.

3.3 Session layer

The scope of the session layer is to offer a synchronization mechanism for entities residing in the layers above.

The design of this layer in HMINET-2 relies on the requirements of the standard applications offered.

The main functionality is a checkpoint/restart mechanisms used for data transfer between peer application entities.

Depending on the reliability class required by the application the frequency of checkpoints is negotiated at the session establishment time.

The main difference between the ISO-OSI documents on session and our design and implementation consists in the perception of the session functionality. If the ISO/SC16 documents are specifying a session layer which offers only a mechanism for synchronization which is triggered by the processes in the layers above, the session layer in HMINET-2 provides the necessary mechanism for reliable communication and synchronization inside the session layer. By this means the applications processes are relieved from communication reliability concerns and synchronization problems.

The evident implication is that data is buffered in the session layer and that the checkpoints refer to session data units which size are session specific and negotiable at the session establishment time. The independence between the session data unit and the application data unit is from our experience an important aspect to achieve the layer independence. The session data unit acts like a container where the application data units are put into for transfer. An application data unit can be transferred using one or more session data units or a session data unit may contain more than one application data units.

The above mechanism has two other important side-aspects which are:

- it offers a flexible mean for resource administration in communication between different hosts. The buffer size for the session data units transfer are negotiable at the session establishment time and are dependent on the communication host types and their load.

- it offers in certain cases a performance improvement if by one session data unit more than one application data are transferred through the network (e.g. file transfer application).

To summarize the implemented mechanism: depending on the application type which initializes the session a certain session data unit size and the checkpoint frequency is proposed to the remote system (session entity).

Depending on the local resources and parameters the remote system accepts or proposes in return its choice. The initiator session entity may now decide if in this conditions the application requirements can be fulfilled or not. In the later case the session is discarded and the application process may decide to establish a new session with decreased quality of service or to try latter in the expectation that the resource set in the session layer will be able to satisfy its requirements.

3.4 Application and presentation layer

The facilities offered by the network can be classified in three categories:

1) standard network facilities conforming the ISO-OSI model (CBMS, virtual file),
2) facilities conforming other standards (terminal access CCITT X3, X28, X29),
3) user software package relying on layer 4 or 5.

We will consider only the first two categories since they are of general interest.

3.4.1 The Computer Based Message System DINAMIT

The DINAMIT system provides four main facilities for the network user:

- mail service, where letters can be sent to one, to a group or to all network users,
- an on line access to a library of documents and help files for the use of the network,
- a network status service where information of general interest about the hosts connected and network status at the service request time is available,
- network management capabilities for the whole network[3].

The system is a fully distributed application where its users may send or receive messages from each terminal connected to the network (i.e. locality independent).

Four types of processes are involved in the system (fig. 5):

- an user agent process on each host from which the DINAMIT system is accessable,
- a post-station process for user message switching. The number of post station processes and their assignment to a certain host is dependent on topological and efficiency criteria.
- a message base for the storage of the messages to be forwarded.
- an user record base for each user of the system, where information about messages sent to him is available (fig. 5).

A message to be sent is delivered by the UAP to the post-station process to which the UAP is connected. The post-station process will store the message in its message base and will inform the addressed user(s) record base(s) about the existence of a message for him in the PS message base. The user(s) for whom the message is destinated will be informed about the existence of a message as soon as he is accessing the DINAMIT system at any of the user agent processes.

The presentation layer functionality is implemented by adopting standard data formats between the communicating processes and by encoding/decoding the standard data representation into the local one.

Note that DINAMIT is a typical example of a distributed application in the ISO-OSI environment, where different distributed processes are involved.

Although the ISO-OSI model describes only a point-to-point type of communication between autonomous application processes of the same type, some comments about the feasibility of the model for applications which are functional distributed are probably interesting:

- there is a need of multicast and broadcast communication type to support such applications. This facility has to be supported by the session layer and below in order to relieve the applications and presentation layer from communication oriented concerns,
- establishing and discarding connections for the transfer of small amounts of data is a very costly and inefficient mechanism. A datagram or fast connect/disconnect service visible in the upper layers will be usefull for a number of applications,
- the synchronization between concurrent accesses to ensure the consistency of distributed information is a service which is mandatory for a large number of functional distributed applications (mail systems, distributed databases etc.). Mechanisms like two-phase commit, time stamps based have to be offered by the session layer.

3.4.2 The virtual file system

The virtual file concept was introduced[10] in order to alleviate the incompatibilities of the different data management system types in networks with a high degree of heterogeneity or for the design of networks with a non-predictable population of hosts (open systems). The concept was adopted in the national and international standardization bodies (ISO, DIN etc.) and was implemented in HMINET and BERNET[8].

The main aim of this approach is to define a standard logical file structure in each host in the network and to provide the different systems with translation mechanisms between the new logical structure and the existing local structures. With this concept, equivalent with the definition of netwide data management system, the user who accesses data somewhere in a network has only one interface; the interface to the virtual file system. The virtual file can be described as a tree structure with n hierarchical levels. Each level i ($1 \le i \le n$) consists of a set of file access data units (FADUs) at level i-1 or below (fig. 6). A file access data unit is an accessable piece of information that has a semantic meaning.

Note that the depth of the tree is open ended and it is user application dependent.

It is obvious that the advantage of introducing an uniform structure is that we can introduce an uniform set of FADU access services and one protocol (i.e. the same services and protocol support file transfer as well as data units access within a virtual file). The depth of the virtual file is negotiable between the two application entities and uniform conventions on authorization, access control etc. are introduced in the network. The common set of services and the related protocol are defined in the nested phases (ISO terminology: regimes):

There are some differences between the implemented virtual file system in HMINET - BERNET environment and the ISO-VF system specification[11].

- the implemented file structure is a subset of the ISO one and consists in the restriction that the arc length between the FADUs is always 1. The depth is negotiable between the systems involved in an application connection.
- for efficiency reasons the implemented system provides the capability of accessing FADUs in different virtual files stored at a certain host within one application connection. The ISO specification provides one application connection per accessed file. The reasoning is that since multiplexing is provided by the transport layer (cost efficiency) then maintaining one application connection per vitual file will not influence the costs. But since the first two classes of the transport protocol are not multiplexing and since there is an inherent sequentiality within most programs accessing data in different files, we decided to provide the mechanism depicted in fig. 7b rather than the ISO one (fig. 7a):
- another difference between the two approaches is that the HMINET-BERNET implementation provides a distinct separa-

tion for recovery for data transfer failures (check-point/restart in the session layer) and the check point /restart mechanism in the application layer related to the accessed structure. In our approach the application entities are not concerned with recovery from failures in the data transfer; the failures being captured within the session layer.

The data types supported by the presentation layer as standard transfer syntax encompass: characters, integers, reals etc. The data type set is defined open-ended and more complex structures for new applications like: graphics, document transfer, remote job entry etc. are in development.

The formal specification of the virtual file system[2] has accompanied the implementation work and has provided the precise basis for the implementation phase in the different computer system types.

The implemented virtual file structure is flexible enough to accommodate the requirements of the traditional file transfer and remote data access (record access) systems and at the same time to map data structures used in applications like graphical systems (metafiles), document access systems etc.

New enhancements of the virtual file protocol for concurrency control have been specified and are in development. The enhanced protocol is based on a time stamp mechanism for a distributed database management system[7].

3.4.3 The terminal access system DIA (CCITT X3, X28, X29)

The decision to support in HMINET-2 the CCITT 3X recommendations was motivated by its national and international dissemination although these standards do not rely on the ISO-OSI model.

The 3X recommendations describe the protocol for start-stop terminals in a X25 network.

The specification was influenced by the initial thoughts that packet-switching networks will support only terminal-host communication and that the network nodes will offer PADs (packet assembly-disassembly) as a common communication carrier service. The initial PAD specification for DCE (data communication equipment) has precipitate the invasion of products like PAD terminal concentrators and implementation of PAD in hosts.

This was motivated by the costs for the PAD/terminal connections and the flexibility in using the same terminal in local mode or as a terminal for a remote host.

Another important aspect to be observed is the fact that the 3X recommendations mainly describe a protocol and that the service definition is poor if not inexistent. The result is that the host behaviour is not specified and is implementation dependent (e.g reset behaviour, echoing etc.).

The DIA system in HMI is offered on each host, in terminal concentrator and in the switching node (fig. 8).

4. Conclusions

There are some aspects for general interest we have learned in the project development:

A) The ISO-OSI model is an efficient tool for the development of the complex hardware and software of a network.

   Adopting the same ISO-OSI reference terminology and design concepts in a large team of network designers and implementators has minimized the understanding overhead.

B) A strict layering architecture in the network design and implementation is mandatory for future changes and integration of new standards and products.

   A network based on the standardization output minimizes in long terms the maintenance and integration work since it can be expected that more and more communication standards will be provided by the computer manufacturers as parts of their operating systems.

C) An active participation in the standardization work has shorted the information path standardization output - implementation.

D) The use of formal description techniques in the specification phase is mandatory in order to achieve compatible implementations by different teams on different machines.

E) The development of testing procedures and reference implementations as certification centers is a necessity for heterogeneous networks and has to be considered as part of the network design from the beginning.

Literature:

1 Bochmann G.v., "Formalized specification of the MLP"
Report Univ. of Montreal June 1979

2 Bochman G.v. et al., "Formalized specification and analysis of the Virtual File System"
HMI Report B367 Feb. 1982

3 Bauerfeld W., et al., "DINAMIT - an adaptable, distributed computer based message system"
to be published

4 ISO/TC97/SC16 Draft International Standard 7498 1982

5 Lenzini L., Popescu-Zeletin R., Vissers C.
"State of the art study on the standardization of level 4" Report for the Commission of European Communities"
Brussel 1981

6 Mier, Edwin E. "High level protocols and the OSI reference model" Data Communication 1982

7 Popescu-Zeletin R., Weber H. "The logical architecture for D-DBMS in the ISO-OSI framework"
Proc. GI Congress Berlin 1983 Springer Verlag

8 Popescu-Zeletin R., et al. "The virtual file specification - final draft - "HMI Report B333 1980

9 Popescu-Zeletin R., Baum D., Butscher B. " A local computer network in a research environment: The HMINET Proc. Computer networking symposium NBS 1979

10 Popescu-Zeletin R., Bauerfeld W. "A virtual file concept for heterogeneous computer networks" Proc. AICA congress BARI Italy 1979

11 ISO/TC97/SC16/WG 5 Working documents on file transfer access and management. N1190, N1222, N1223, N1224 revised 1983

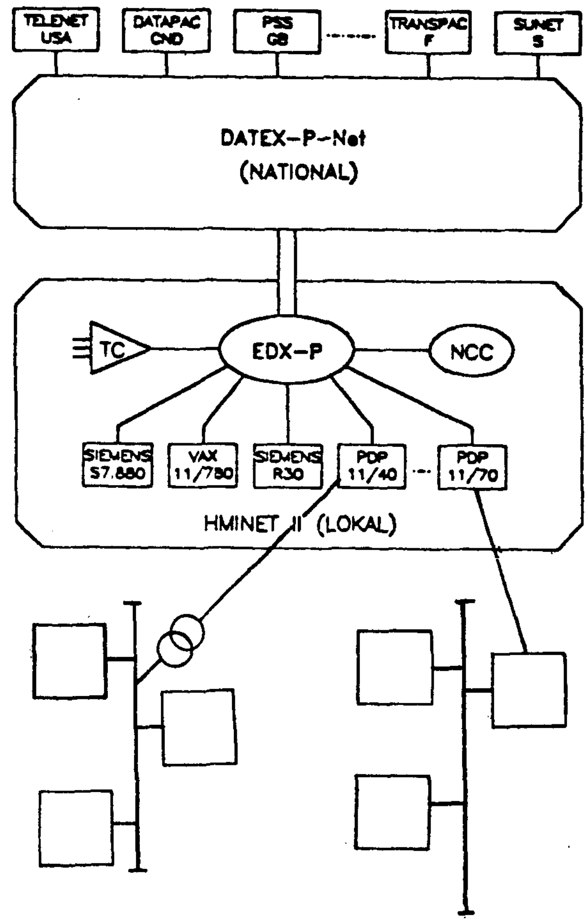


Fig.1

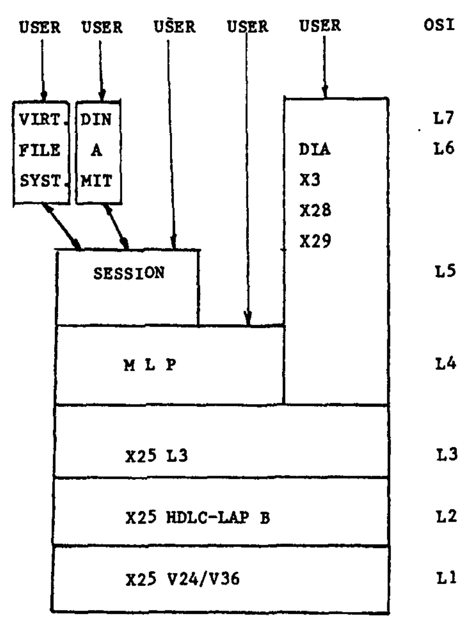


Fig.2

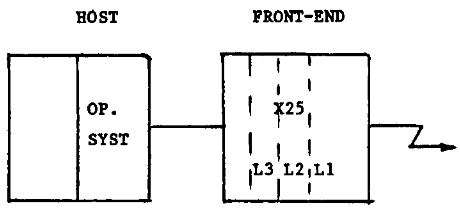


Fig.3a

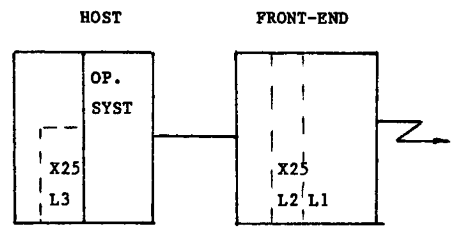


Fig.3b

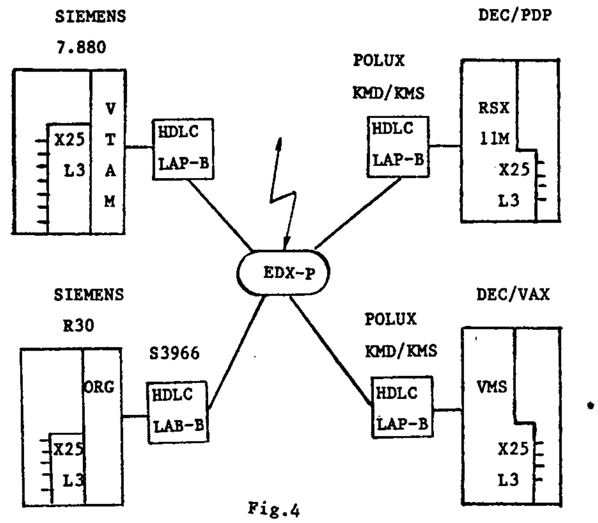


Fig.4

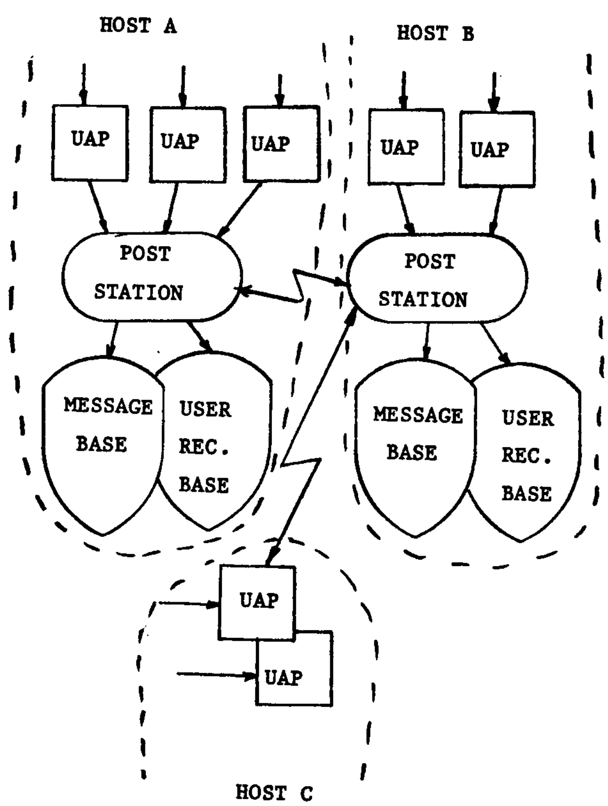
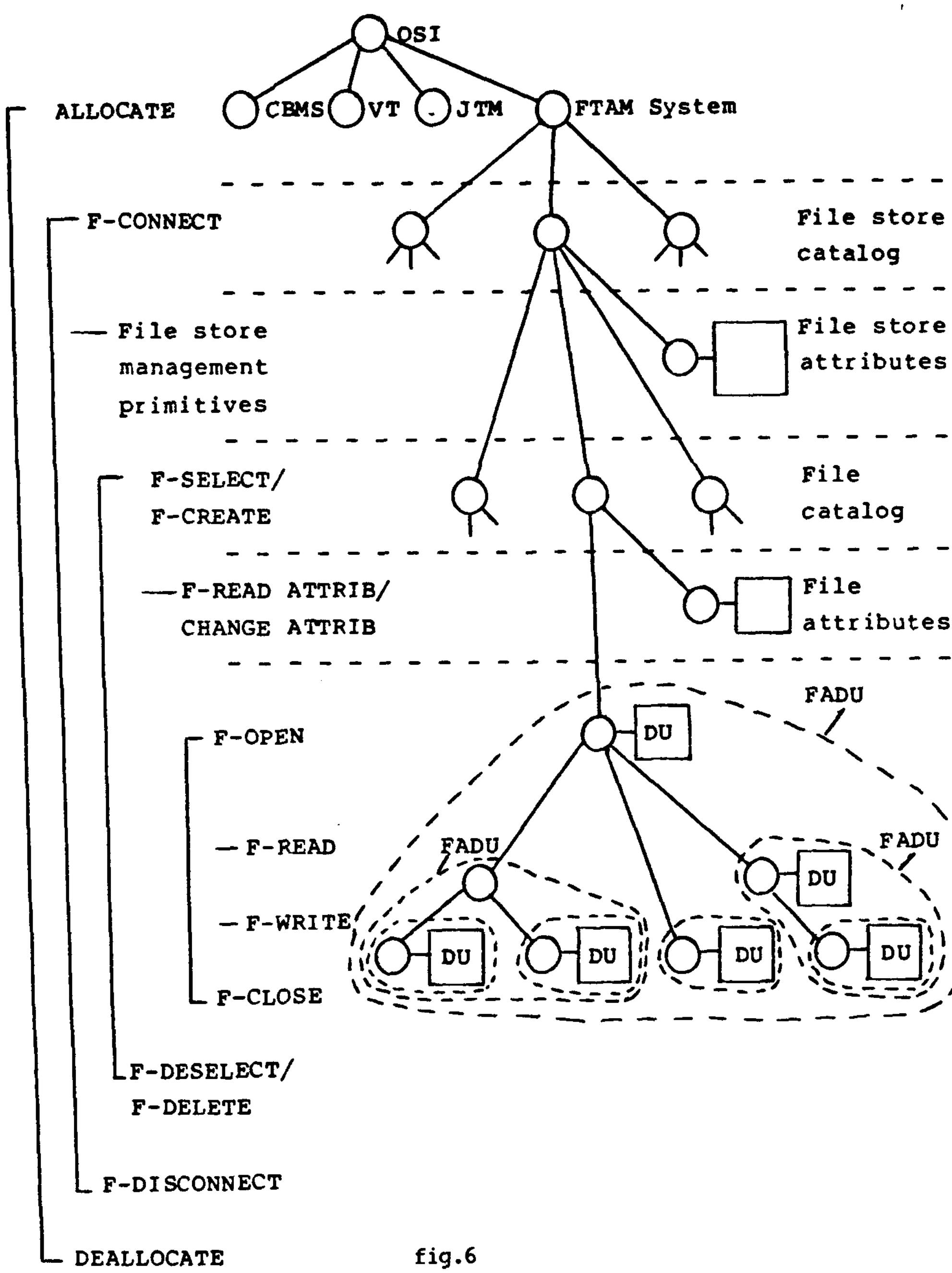


Fig.5


OSI
ALLOCATE
CBMS
VT
JTM
FTAM System
F-CONNECT
File store catalog
File store management primitives
File store attributes
F-SELECT/ F-CREATE
File catalog
F-READ ATTRIB/ CHANGE ATTRIB
File attributes
FADU
F-OPEN
DU
F-READ
FADU
FADU
F-WRITE
DU
DU
DU
DU
DU
F-CLOSE
F-DESELECT/ F-DELETE
F-DISCONNECT
DEALLOCATE
fig.6

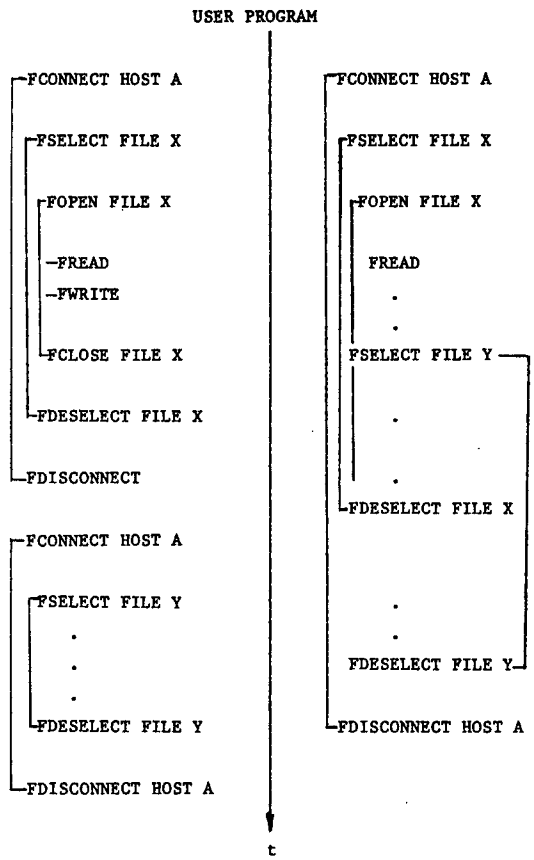


Fig.7a

Fig.7b

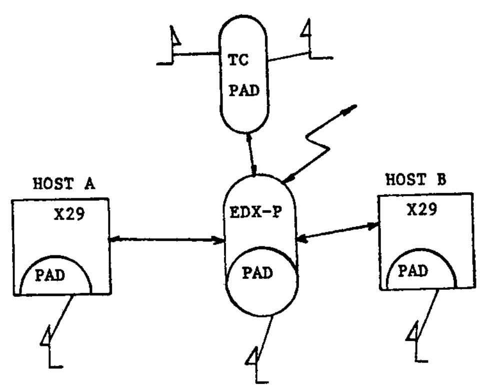


Fig.8