Open System Interconnection: Application Issues Associated with the ISO and CCITT Layered Models

MICROPROCESSOR SYSTEMS DEVELOPMENT LABORATORY
DEPARTMENT OF SYSTEMS ENGINEERING AND COMPUTING SCIENCE
CARLETON UNIVERSITY

Ours The Task Eternal

Microprocessor Systems Development Laboratory
Department of Systems Engineenring and Computing Science
Carleton University

MSDL Tech Note 79-1 (DOC)
March 31, 1980

Open System Interconnection: Application Issues Associated with the ISO and CCITT Layered Models

by

R.J.A. Buhr - Principal Investigator
S.A. Mahmoud - Associate Researcher
K. Toth - Graduate Student

Prepared for the Department of Communications
under Contract OSU79-00173
"Open System Architecture"

1. Introduction

1.1 Background

This report presents the results of a study performed by Carleton University under a DOC University Research Contract entitled "Open System Architecture".

The report is a contribution to the Canadian Government's efforts to gain insight into the problems of developing unified Canadian standards for multipurpose terminals capable of interworking over a variety of national and international network services.

It is important to the ultimate success of the Canadian electronic/communications industry in the international marketplace that intelligent terminals capable of communicating with remote computers and remote intelligent terminals do so in accordance with emerging international standards for communication protocols. And it is equally important that Canadian development efforts already completed or under way in the intelligent terminals area have some influence on these emerging standards. In some cases these developments may even serve as models for the standards.

Of particular concern is the problem of interconnecting intelligent terminals and computers of different types and with different, but complementary, functions using public networks of various descriptions. Emerging ISO and CCITT standards for open system interconnection are aimed at providing solutions to this problem by dividing the communication interface into seven standard levels with standard services and functions at each level.

The levels and a very rough description of their functions are as follows:

1. The physical level is concerned with electrical connections.

2. The link level is concerned with control of data transfer over a physical connection.

3. The network level is concerned with access and use of the services of a public network.

4. The Transport level unifies the services of different public networks into a standard service interface for all networks (although there is some controversy over this level).

5. The Session level provides a virtualized transmission path for application-related messages which have been formatted for communications by the presentation level.

6. The Presentation level controls the formatting/-transformation of application-level data units into communications- level data units and vice versa.

7. The Application level handles strictly application-oriented functions not handled by the lower levels.

Each level may have recovery and other control functions appropriate to that level. Standards conforming to the Open System models exist for particular types of network services for Levels 1-3. There are no such agreed standards, but only draft recommendations, at all levels from 4 to 7.

Intelligent terminal types of concern to this study are as follows:

1. Word processors.
2. Tradex terminals.
3. Electronic mail stations.
4. Teletex terminals.
5. Videotex terminals.
6. Facsimile terminals.

Of concern is the capability for interworking both among terminals of a similar type and among terminals of different types. Also of concern is the possibility of the emergence of multi-function terminals combining some of these types. Combined Tradex/Word-Processing terminals are in development now. Combined Word-Processing/Teletex terminals are a possibility. Other combinations are easily visualized.

The only two types commercially available to date in Canada are Word Processors and Facsimile terminals and the current level of capability for interworking is very limited, even among terminals of a similar type. Usually it is restricted to terminals from the same manufacturer. Some limited intercommunication capability between some word processors from different manufacturers is possible, but with very restricted protocols and not without some loss of text formatting information. These terminals were designed and developed before Open System Interconnection models began emerging and could not, therefore, benefit from these models.

The remaining terminal types are currently in various stages of prototype development and/or test in Canada. These activities are taking place concurrently with the development of Open System Interconnection standards. It is therefore inevitable that mismatches occur between the prototype designs and the emerging standards.

The first Canadian prototype Tradex terminal was demonstrated in May 78 and a limited production version is scheduled for release in mid-1980. Neither of these versions is in detailed conformance with emerging Open System Interconnection standards except for the Network Level in the X.25 version. Efforts are being made by COSTPRO, apparently with some success, to achieve international standardization on trade data formats at the Application Level. However, this is a slow process and the only standard adhered to at the application level by the soon-to-be-released Tradex terminal is that of the Canadian Trade Document Alignment System (CTDAS).

Electronic mail experiments are being conducted in various places in Canada. TCTS offers a store-and-forward message service over Datapac. And experiments are under way using word processors as electronic mail terminals. No specific international standardization activity for electronic mail is known to the authors. However there has apparently been discussion of using Videotex as a basis for Electronic Mail. A Canadian version of the Videotex terminal (called Telidon) is presently undergoing demonstration and test. Little specific provision has been made in its design for accommodating Open System Interconnection standards, although some thinking is now taking place in CCITT on how to do it. There has apparently been some thought in the Videotex study group that Videotex could be used as the basis for a wide variety of services.

A current effort by CP Telecommunications is expected to result soon in the availability of a version of Teletex (a sort of "super-Telex") called Infotext. The Infotext effort is being used as a basis for the drafting of Teletex standards and, as a result, fairly detailed draft recommendations for some of the levels of the Teletex service became available recently. These draft recommendations are broadly in conformance with the Open System Interconnection models.

The specific experience the authors of this report bring to bear on the issues is their involvement with several projects of Carleton's Microprocessor Systems Development Laboratory which successfully implemented intelligent terminal systems in the TRADEX and Electronic Mail areas. A significant part of this effort has been involved with the adaptation of existing Word Processing terminals for these functions.

In addition to this practical experience, a parallel study for the Department of Communications involving one of the authors (Buhr) under a Research Contract entitled "Word Teleprocessing Interface" investigated problems associated with communicating word processors which are of some relevance to this study.

This report presents the views of a team of designers and implementors of intelligent terminals and protocols on the main practical, applications-oriented issues in implementing Open System Interconnection standards and models. The authors sit on no standardization committees or bodies and have not been privy to the discussions of issues which have taken place in these bodies. And it was outside the terms of reference of this project to interview standardization experts in CCITT and ISO to obtain their views. Views were sought informally in a number of cases, but this report makes no claim to represent these views. Accordingly the authors claim no status as standardization experts per se. Their views, rather, are those of potential users of standards. As such, their views may assist those standardization experts who do sit on the standardization committees in judging the practical importance of certain issues.

1.2 Terms of Reference and Scope of the Study

The terms of reference are best stated by quoting Sections 5 and 7 of the project proposal (which is part of the contract):

"5. Purpose of Research

Objectives

The main objectives of this study are as follows:

1. To evaluate the practicality of the ISO/CCITT concepts and ideas for a layered model of open system architecture relative to a number of practical applications.

2. To identify any problems which may exist in fitting the applications within the architecture without disturbing the latter's logical structure.

3. To make inferences about the logical soundness of the architecture concepts and ideas, including criteria for partitioning the layers, the entities within a layer and the protocols and interfaces between and among layers.

Because the application areas to be studied are themselves evolving and not yet fully defined, an important sub-objective is to define as precisely as possible the relevant requirements of these application areas, based on available reports and papers and on the particular experience of Carleton in the Electronic Mail application area.

Scope

In order to make progress within the time and budget constraints, the study will be restricted in scope as follows:

Only a small number of applications will be studied; the four mentioned in the introduction are proposed. Furthermore the study will not set out to make specific proposals for changes to the open system architecture specifications; rather it will aim at providing input to the process of making such changes (if required).

Because a major input to the study will be the relevant experience of Carleton in the Electronic Mail area with COSTPRO and AES, it is expected that this area will be emphasized in the study of applications."

"7. Brief Description of Research Method to be Used

Statement of Work

The project will be conducted in five phases as follows:

Phase 1: Information gathering
Phase 2: Preliminary evaluation of open system architecture applied to communicating word processors in an electronic mail environment.
Phase 3: Development of a consistent description of the interworking functions and procedures required for the range of applications being studied.
Phase 4: Final evaluation of open system architecture in terms of the interworking functions described in Phase 3.
Phase 5: Development of inferences, conclusions and recommendations.

During Phase 1 Carleton will assemble and review all relevant documentation to be used in the study. Sources will be primarily DOC with some material from AES and some material from Carleton's own internal notes and files. A brief report on Phase 1, listing all source material, will be provided to DOC by 1 September 1979.

Due to previous commitments by Carleton, intensive work on Phase 2 and subsequent phases will not start until 1 September 1979.

Phase 2 will be used to develop a solid foundation for the remainder of the study, based on Carleton's existing experience. Work in this phase will concentrate on making a preliminary identification of any problem areas with the layered model on developing a methodology for describing the functions and procedures required for interworking and on developing a detailed plan for Phases 3 and 4. A report on Phase 2 will be provided to DOC by 1 December 1979.
Phases 3 and 4 will proceed according to the plan developed during Phase 2. Reports will be submitted on these phases by 1 January 1980 and 1 March 1980, respectively.

Phase 5 will be completed and the final report submitted by 1 April 1980."

A substantial portion of the work was to be based on MSDL experience in designing and implementing intelligent terminals containing a variety of Network Level protocols, including X.25, and in studying the requirements of communicating word processors. While this work has no status as a standardization activity, it was thought to be relevant to the aims of this study. Accordingly it was proposed to devote part of the study report to a description of the work and to the issues arising from it, as a means of identifying key open system interconnection issues as seen from the applications point of view.

At the time the contract was negotiated the only report in the hands of the authors on Open Systems Interconnection was ISO/TC97/SC16/N227 dated August 79. It was planned at the time to relate this report to existing CCITT and MSDL work in the specific application areas. Information on CCITT work in the specific application areas was to have been gathered in the Phase 1 of the project. The original target date for completion of Phase 1 was 1 Sept 1979. However, for a variety of administrative reasons within both Carleton and the government, the contract was not issued until September 79 and the project did not actually start until November. Therefore the project schedule was compressed and Phase 1 had to proceed in parallel with the analysis phases during a very active period for the emergence of new draft standards from CCITT, both for Open System Interconnection and for Teletex. As these standards emerged, the scope of the project was broadened informally to include them.

As well, it became apparent as the work progressed that the possibility of multi-function terminals was of major interest and, at the request of the Scientific Authority, the scope was informally broadened to include them.

1.3 Outline of the Report

Chapter 2 provides a brief introduction to the Open System Interconnection models. It describes the methodology to be used throughout the report in describing the services and functions of the levels in the models for various applications. And it comments on some aspects of the clarity and comprehensibility of the specifications for the models.

Chapter 3 is based on MSDL experience in developing Tradex and experimental Electronic Mail Systems and in studying the requirements of Communicating Word Processors. It describes the main relevant features of Tradex and Electronic Mail prototypes implemented by the MSDL. It comments on Open System Interconnection issues now apparent to the designers with the benefit both of hindsight and of a deeper understanding of the Open System Interconnection models. (The only Open System model documentation available during the development of the software architecture of the terminals was an early, very brief and incomplete description of the ISO Model which was used as a guide only). Finally it discusses Open System Interconnection issues associated with Communicating Word Processors.

Chapter 4 comments on the draft Teletex standards primarily as they relate to the possibility of including Teletex in a multi-function terminal.

Chapter 5 discusses issues for Facsimile terminals in Open System Interconnection and provides a conceptual proposal of how they can fit within the open system model, provided the model is interpreted somewhat liberally.

Chapter 6 discusses issues for Videotex terminals in Open System Interconnection, with some reference to Telidon.

Chapter 7 summarizes the issues by application and level and comments on the implications for multipurpose terminals.

Chapter 8 presents conclusions, and recommendations for further study. As appropriate to the intent and scope of the study, no specific recommendations with respect to the wording of draft standards are presented.

2. Open System Interconnection Models

2.1 Introduction

The seven levels of open system interconnection are shown in Figure 2.1. They have been roughly described in Chapter 1. They are described in much greater detail in two reports by the ISO [1] and CCITT [2]. These two reports correspond exactly in the names and numbers of levels, but otherwise differ significantly. They use different terminology and different philosophical approaches, so that it is often difficult to be sure whether apparent differences are real. The situation is further complicated by the fact that each report is itself unclear in many respects (see Section 2.2).

It is not the purpose of this report to make a detailed comparison between the ISO and CCITT models. The terminology differences and unclear aspects of both reports would make such a detailed comparison by the present authors of dubious validity.

Instead, this report takes the ISO report as its philosophical base and concentrates on the question of what should the services and functions of the levels be to accommodate the multiple applications which the authors have investigated.

Rather than describing each level in detail in this chapter, the report disucsses each level for each application in Chapters 3-6 and then draws the results together for the range of applications in Chapter 7.

2.2 Methodology

In general, the ISO and CCITT reports and draft recommendations lack clarity with respect to the interfaces between levels. It is often not clear whether certain internal functions and procedures of a level are to be performed autonomously within that level, based on the state of the level itself, or are to be performed only as a direct result of a service request from a higher level. This is particularly true of the handling of protocol messages. The protocol messages of a level are often described in considerable detail but their relationship to service requests of that level is often left until the internal functions and procedures of higher levels are described. Since the recommendations are often defined from the bottom up and, in draft form, are often missing levels, the net effect is often unclear. The effect is confusing even when all levels are completely defined, because each level cannot be understood in isolation.

It is also often unclear whether some services and functions of a level are to be performed directly by that level or by service request to a lower level. Examples will be cited in subsequent chapters. In this chapter we wish only to present some conventions for describing interfaces which we have found to be useful in clarifying these issues. These conventions are used throughout this report.

Figures 2.2 and 2.3 show graphical conventions to represent services and functions of a level which aid in visualizing the purpose of the level both in isolation from and in relation to other levels.

Figure 2.2(a) shows how level services are represented in isolation. Services are provided by service primitives represented by labelled arrows drawn to a box representing the level. For example, a request to establish a session with a remote terminal would be a service primitive of the session level; so would a request to wait or check for a remotely-requested session. Our view is that services of a level are only accessible from above. No service requests are ever made of a level from below. Instead, autonomously arriving remote requests to a level arrive in messages to that level from below. And they are only accepted by that level if the appropriate service is active. For example, requests from remote nodes to establish sessions are handled by the "wait-for-a-session" service of the session level, which is activated by service requests from higher levels.

Figure 2.2(b) depicts the case where a service of a level is actually provided by a service of a lower level. For example, a "place call" service of the transport level might be provided directly by the "place call" service of the Network level for an X.25 network.

Figure 2.2(c) shows internal use by a level of a lower level service. For example, the presentation level might use a session level service to send a message to a remote presentation level to obtain permission to send a document or file.

Figure 2.3(a), (b) and (c) show how messages passed on by other levels may be distinguished from those generated within a level as a direct result of a service request and from those generated autonomously within a level as result only of the state of the level. For example, presentation level data or control messages would be simply passed on by the session level as shown by the downward arrow in Figure 2.3(a).

A service request to the session level to establish a session would result in a session establishment request message being generated by the session level and passed on to the lower level, with a positive or negative response message eventually passed back by the lower level to the interface, as shown in Figure 2.3(b). And internally, session control messages relating to recovery might be generated autonomously as shown in Figure 2.2(c), with the responses hidden internally or passed on to the interface as shown in Figure 2.3(c).

In general, functions of a level are described satisfactorily by finite state machines (FSMs) which govern the operation of the protocols of the level. This approach will be used throughout this report.

It is often stated in the context that formal protocol specification techniques are required. While formal techniques are desirable, the problems with lack of clarity and ambiguity we have observed would be alleviated in large measure if consistent English-language and pictorial representations were used to define interfaces, functions and message flows. And if strong attempts were made to distinguish interfaces, external services, and internal functions.

2.3 References for Chapter 2

1. ISO/TC97/SC16/N227, Aug 79, "Reference Model of Open System Interconnection (Version 4 as of June 79)".

2. CCITT/Study Group 8/COM VIII - No. 394E, October 79, "Draft New Question VII/XX - Structure for and Use of a Reference Model for Public Data Network Applications".

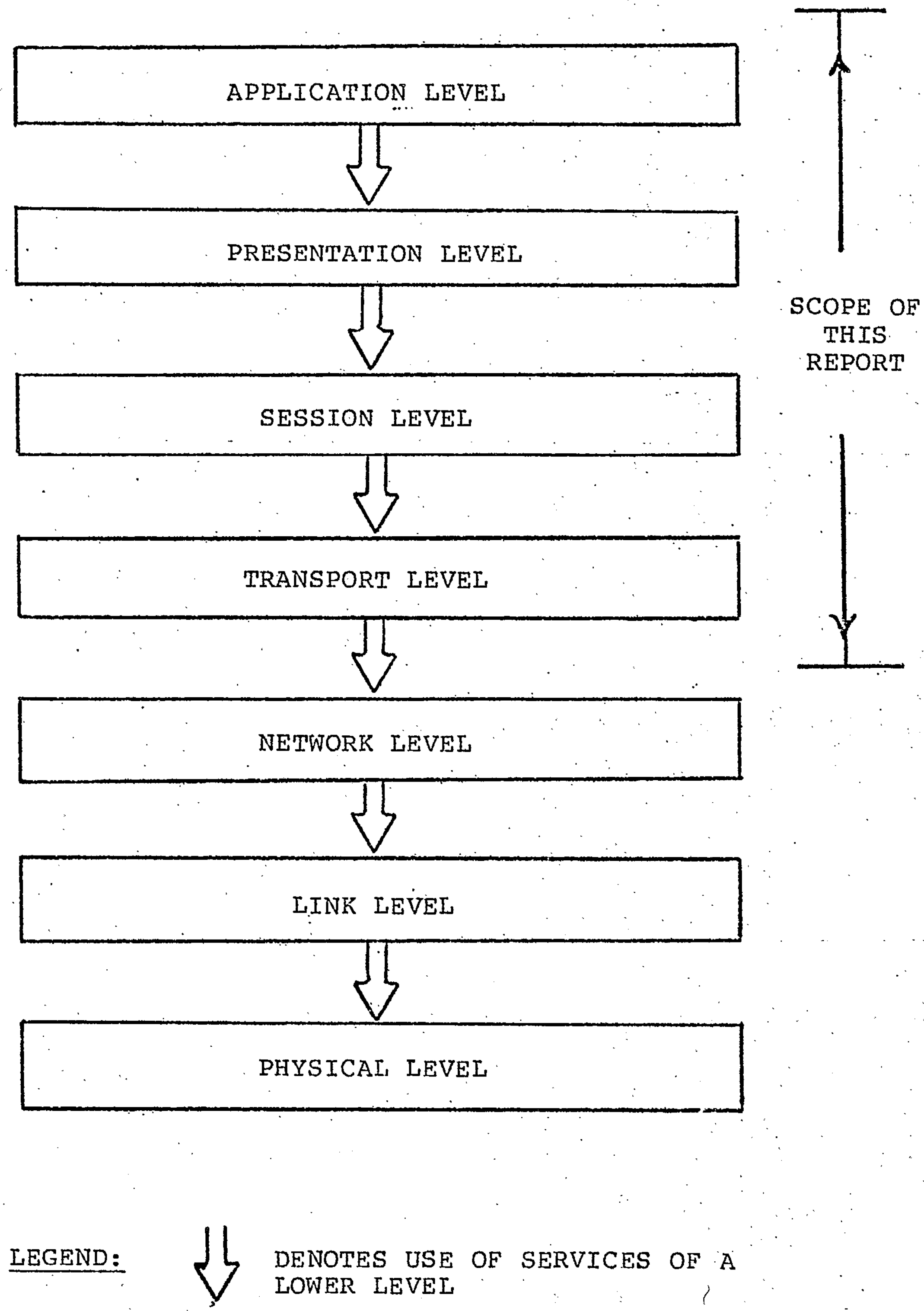


Figure 2.1: The Layered Model

(a)

SERVICES OF A LEVEL

(b)

SERVICES OF A LEVEL PROVIDED BY A LOWER LEVEL

(b)

SERVICES OF A LEVEL PROVIDED BY A LOWER LEVEL

(c)

INTERNAL USE OF LOWER LEVEL SERVICES

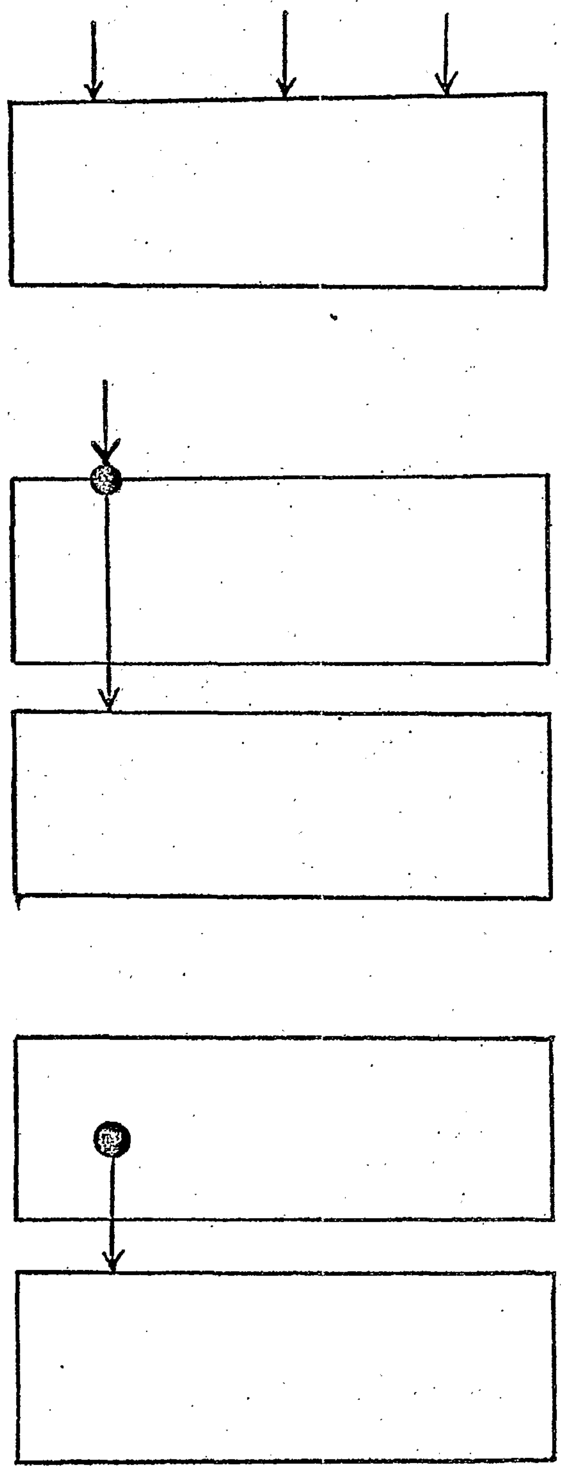
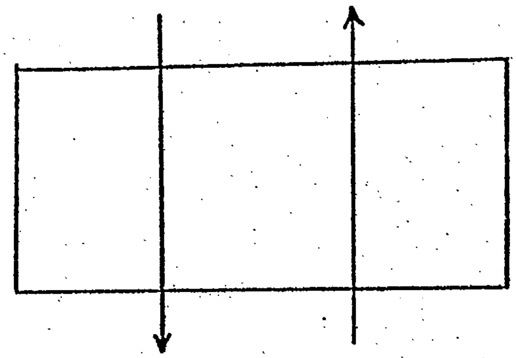


Figure 2.2: Conventions for Representing Services of Levels

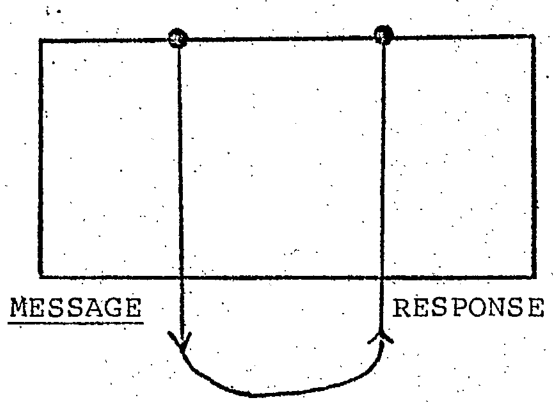
(a)

Messages passed on to Higher or Lower Levels



(b)

Messages Originating/ Terminating in Service Requests at the Level Interface



(c)

Messages Originating/ Terminating Internally in the Level

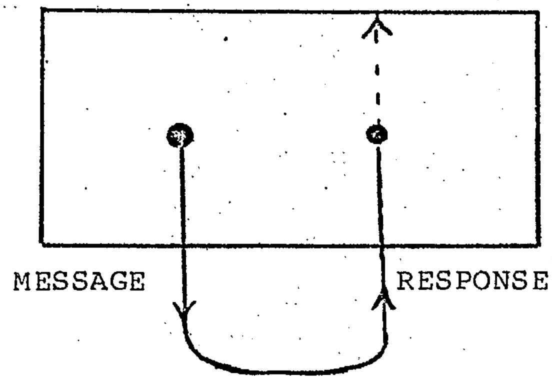


Figure 2.3: Conventions for Representing Message Flows Between Levels

3 TRADEX, Electronic Mail and Communicating Word Processor Applications: Layered Model Issues arising from MSDL Experience

3.1 Introduction

This chapter draws on MSDL experience with three network applications to identify key issues with respect to the services and functions of the top four levels of the ISO and CCITT Models of Open System Interconnection.

The three applications are TRADEX, Electronic Mail and Communicating Word Processors. The MSDL has designed and implemented several actual systems for the first two applications. The basis for MSDL experience in the third application is twofold: some of the work performed on the first two applications has been based on existing word processing terminals; and a study has been conducted for DOC in parallel with this project to identify problems and potential solutions associated with communicating word processors.

None of the MSDL work in these three application areas has any formal status with respect to international standardization activities. Nor did it explicitly attempt to satisfy emerging open system interconnection standards above the network level, except on an informal basis. Nevertheless, the MSDL experience provides a relevant basis for commenting on issues with respect to these levels.

This chapter is organized as follows:

Section 3.2 describes the general features of the TRADEX, Electronic Mail and Communicating Word Processor Applications from a high level viewpoint.

Sections 3.3, 3.4, 3.5 and 3.6 discuss, from the bottom-up, the services, functions and issues associated with each of the levels of the models for each of the applications.

The sequence is bottom-up because, except for the Transport Level, the open system model levels are progressively less clearly defined as one moves from the bottom to the top levels. Starting at the bottom enables the unclear issues to be gradually introduced in a step-by-step fashion.

Section 3.3 covers the Transport and lower levels. Section 3.4 covers the Session Level. Section 3.5 covers the Presentation level. Section 3.6 covers the Application level.
Finally Section 3.7 summarizes the issues and Section 3.8 lists the references.

3.2 Description of the Applications

3.2.1 Introduction

In this chapter, issues associated with layered communications in TRADEX, Electronic Mail and Communicating Word Processor applications are discussed, based on MSDL experience with these applications. These applications are described from a high level viewpoint in Sections 3.2.2, 3.2.3 and 3.2.4. For concreteness, subsequent sections of this chapter describe and discuss specific services, protocols and functions of the various layers of prototype systems designed for the first two applications. It should be emphasized that these descriptions provide a static and incomplete view of an evolving set of system designs. No inference should be drawn that they describe correctly the details of any particular final system.

3.2.2 TRADEX (Trade Data Exchange)

Requirements for the TRADEX terminal have been specified by COSTPRO (Canadian Organization for Simplification of Trade Procedures) in Reference [1]. Several prototype versions of TRADEX terminals have been designed and implemented by the MSDL. Many of the issues associated with Open System Interconnection models have been resolved in practice in particular ways in these projects. An important exception is code and format conversions to accommodate unlike TRADEX terminals with unlike data file formats. All MSDL TRADEX projects have assumed like terminals and like data file formats, for the simple reason that during the design stages of these projects, no compatible standards were available which would have allowed greater generality. However the issue of unlike terminals and file formats did arise on many occasions and will accordingly receive comment.

Figure 3.1 depicts a TRADEX environment as seen by the human operator. Operators may send messages from the screen or files from the disk directly to a remote TRADEX terminal. Files may be sent singly or batched. File names have local significance only. Operators must be directly involved with file naming, storage, retrieval, copying, etc. Files may be trade documents, trade document formats or ordinary text.

In a TRADEX-type environment it is necessary to distinguish at the human operator level between two different views of communications. In one view, sessions between terminals are operator-initiated. In the other view they are system-initiated.

With the first view, a session is established by the operators between terminals at different network nodes, allows the operators to interchange data using their CRT screens or file systems. Operators are aware of the session concept; indeed, they must explicitly request that sessions be established. Once a session has been established it may be used by the operator at either end to send messages from the screen or files from disk. Arriving messages are displayed directly to the operator and then filed. Arriving files are stored on disk under an appropriate local file name. The session remains established until specifically cleared by the operator at either end. Indeed it may remain established even while nothing is happening.

An example of the type of operation is as follows (short messages are in quotes):

Sending Operator S	Receiving Operator R
- Requests session with R	
- Session established	- Session established
- Sends "I have an urgent file for you"	
	- Receives "I have an urgent file for you" - Sends "Send it" - Waits
- Receives "Send it"	
- Sends file	

Sending Operator	Receiving Operator
	- File stored on disk - Operator picks up file from incoming queue - Sends "I got it"
- Receives "I got it"	- Proceeds to take action on urgent file
- Clears session	
- Session cleared	- Session cleared

An operator-initiated session can also be used simply to send a file, as follows:

Sending Operator	Receiving Operator
- Requests session - Session established - Sends file	- Session established
- Clears session - Session cleared	- Session cleared - Time passes - Operator checks receive log and picks up file

In the second view of communications, operators are not specifically aware of sessions. Instead sessions are established automatically by the application layer software as required for transmission and reception of messages and files. There is no operator-to-operator dialogue. Instead, files and messages are sent as follows:

Sending Operator	Receiving Operator
- Requests file or message be sent	- Checks receive log periodically and picks up file or message

In general both types of operations may be available on the same TRADEX terminal by arranging that both session establishment and file transfer commands are available to the operator. A file transfer command, without prior session establishment, results automatically in a session being established and then cleared when the file is sent. Otherwise the operator directly controls session establishment and clearing.

3.2.3 Electronic Mail

A particular view of electronic mail is presented here based on MSDL experience with an electronic mail experiment. As with the TRADEX application the assumption is that mail stations are implemented using like terminals with like file formats.

Figure 3.2 depicts a particular Electronic Mail environment as seen by the human operator. In this environment the operator deals with mail files as mail rather than files. The system handles all details of mail file queueing, naming, storage, retrieval, copying, etc. This is accomplished by associating with each item of mail an envelope page which contains much the same type of information as the physical envelope in the existing manual mail system. Details of the envelope page are not important here. Suffice it to say that it provides for sender and receiver identification, privacy and urgency levels, unique mail item identification and registered mail.

Users post and receive their mail via private diskettes which are brought to the mail terminal. They are not aware of mail sessions or network concepts. All mail is batched by the system and transmitted at periodic intervals automatically. The private diskettes contain standard page-oriented word processing files for each item of mail. Registered mail is handled automatically by the receiving system, which simply sends back the envelope page of the registered mail item to the sender under the cover of a new envelope page. Thus the registered mail acknowledgement is handled simply as another item of mail.

Privacy and urgency functions are handled strictly locally at the application level.

3.2.4 Communicating Word Processors

Incompatibilities between different types of word processors from different manufacturers make the successful transfer of editable files either very difficult or impossible. The raw text is not the major problem, though even here problems do arise because of special text characters used by different manufacturers for different markets. The major problems are associated with special control and formatting information and with non-standard or limited communications protocols used by different manufacturers.

A companion report [2] describes these problems and identifies potential solutions. Here we only comment on the issues with respect to open system interconnection standards.

Obviously, communicating word processors could use the file-transfer method of operation or the mail method of operation. The issues arising for those two methods of operation from MSDL Tradex and Electronic Mail projects are also relevant to communicating word processors.

The main separate issue arising specifically for communicating word processors is that of unlike terminals and data file formats. The companion report discusses these issues in detail. In this chapter we are mainly concerned with how solutions can fit into the open system interconnection framework.

3.3 Transport and Lower Levels

3.3.1 Introduction

The transport level in MSDL systems provides services to perform the following functions:

- Place, receive and clear calls

- Send and receive messages of variable length

In MSDL application these services have been provided in a variety of projects using both Datapac (an X.25 service) and Asynchronous Infocall (an X.21-like service). The general framework is shown in Figure 3.3. The message service is generally available. The optional direct packet service is available only for packet networks and simply provides direct access to the network layer services of the packet network.

The transport layer in the MSDL applications is typically just a procedural interface to the network layer which provides a standard view of different networks, as shown by Figure 3.3. It generates no autonomous message activity of its own. For example, for a packet network the send message service is typically a simple procedure which breaks up the message into packets and places successive calls on the send packet service of the network layer. The only waiting involved is for the packet level flow control window to open again if it is closed.

The functions of the X.25 network level are implemented using control and information packets and are coordinated through state variables in the packet monitor. Details are outside the scope of this report. Interested readers are referred to [3] and [4] for further details.

3.3.2 Issues at the Transport and Lower Levels

The only issues at the transport level which have arisen in MSDL systems have been

- network independence
- end-to-end acknowledgement

Duplication of function at the session level to achieve network independence is a potential problem when two such diverse services as Datapac and Asynchronous Infocall are both to be accessed. Establishing a call at the network level in the latter service does not include provision of the caller's identity to the called terminal. Additional end-to-end information must be exchanged over the established call to provide the caller's identity. Should this be done by the network level's place call service? Should it be done by the corresponding transport level service? Or should it be done by the session level "request start session" service?

The latter solution was adopted for simplicity in one Tradex prototype system with an X.21-like network service. The result is duplication of the functions at the session and network levels when an X.25 network is used, if the session level is to be the same for both networks. It might be better to give the Transport level this responsibility, in line with its function of providing a standard view of different networks.

A second issue at the transport level which arose in MSDL implementations has been that of end-to-end acknowledgements in the transport level message service. In one of the MSDL's first systems using X.25, transport level messages were numbered; gaps or duplications in the message numbering sequence at the receiving end caused the transport level at that end to clear the call and report the error to the next higher level. However, such errors presumably occur very infrequently in practice and in subsequent systems this sequence check was judged unnecessary. We now consider that end-to-end checks and acknowledgements of any kind can be safely left to higher levels.

One might ask the question: If the Transport level's functions are as easily taken over by higher levels, is the Transport Level necessary? This is an open question to the authors and is also apparently still an open question within the CCITT.

Issues at the lower levels are outside the scope of this report.

3.4 Session Level

3.4.1 Introduction

Our view of the session concept is that it is a generalization of the lower level concept of a "call", as depicted in Figure 3.6 and described below.

A "call" is established by the network control level's place call service. The result of call establishment is the availability of a channel between network nodes for the transport of data. Messages from higher levels are blocked as required for the network control level by the transport level. The higher levels see only the concepts of call and message. They view the transport level as providing a message channel between nodes.

However, the functionality associated with the "call" concept is limited. It is up to levels above the network level (not necessarily just to the session level) to perform functions such as

- distinguishing messages related to different applications,
- validating passwords and authenticating senders,
- gathering communications statistics associated with different applications (perhaps for billing purposes),
- recovering from message sequence failures (duplications, gaps, timeouts) for different applications, separately,
- performing flow control for different applications, separately,
- checkpointing different applications, separately (checking on status of reception of a sequence of messages), and
- placing and clearing calls as appropriate for each application, separately.

Accordingly, it is natural to introduce the concept of a session which generalizes the concept of a "call" from that of a message channel between nodes to that of a message channel between applications at the nodes. The session layer may include some or all of the functions enumerated above for each session; at issue is, which ones?

A set of general session level services is shown in Figure 3.7.

Corresponding to these services are the session level messages shown in Figure 3.8. The session level acts as a conduit for messages to and from higher levels as well as generating its own messages. In the basic session level, outgoing messages originating inside the session level are sent only as a direct consequence of service requests (solid arrows in Figure 3.8). Only the status request service requires explanation. It could be used to change the mode of operation of the session, as might be required in the Tradex system to change from an operator-initiated to a system-initiated session service in one session. Or it could be used to query the status of an inexplicably inactive session.

A more general session level could provide the same service requests with greater functionality by adding autonomously generated messages for status confirmation and session reset control (dotted arrows in Figure 3.8). For example, prolonged inactivity could trigger the session level to send autonomously a session status request. And gaps in session message numbers could trigger the session level to send autonomously a session reset request. An issue is whether or not the session level should include these functions.

Candidate fields for session level messages are listed in Figure 3.9. The basic fields are probably required in any session level (although not every field is required in all messages). The only fields requiring explanation are the "cause" and "calling terminal ID" fields. The "cause" field is useful for inclusion in session reset request, session start deny and session clear request messages to indicate the reason for the message. The "calling terminal ID" field may be used for the caller's telephone number when interfacing with an X.21-like network service. The optional fields are required in certain messages for some applications. The status and recovery fields are required to provide the additional internal functionality indicated by the dotted arrows in Figure 3.8.

Issues arising at the session layer are related to

- application independence and
- level of functionality

These issues are discussed in Sections 3.4.2 and 3.4.3 based on MSDL implementations of Tradex and Electronic Mail systems and in Section 3.4.4 based on MSDL studies of Communicating Word Processors.

3.4.2 Sessions in the TRADEX Environment

In the first TRADEX prototype, sessions are deliberately quite limited in functionality, for the following reasons:

1. Memory limitations dictated the simplest possible system consistent with the required level of functionality.

2. Open system interconnection specifications were not sufficiently well defined at the time of the first design freeze to make a commitment to a more general session level.

3. A requirement for open system interconnection was not specified.

4. Sessions themselves are very simple (only one session outstanding at a time, only one fixed application, no session billing).

Therefore the session level is used only to establish the modes of operation of the connected terminals and (when an X.21-like service is used) to provide the identity of the calling terminal. The session level services required to do this are "session start", "session wait", "session clear" and "session status request". The session level send and receive message services are provided directly by the transport level. The message fields required are only a small subset of Figure 3.9, namely only the message type, message data and calling terminal ID fields.

The resulting session establishment FSM is shown in Figure 3.10 for interfacing to an X.21-like service. Clearly there is redundancy in this protocol if the network level is an X.25 service.

As discussed in Section 3.2.2, sessions may be visible to or hidden from the human operator. This raises the issue of the application independence of the session level with respect to the placing and clearing of calls. Possibilities are illustrated in Figure 3.11.

In Figure 3.11(a) a single session and a single call both may span periods of session inactivity. Sessions and calls are in one-to-one correspondence.

In Figure 3.11(b) the session remains established during periods of inactivity but calls are cleared during such periods (either after operator-requested activity is completed or after a time-out period with no activity). The session may be said to be temporarily suspended during this period.

In Figure 3.11(c) both the session and the call are cleared during periods of inactivity. Sessions and calls are in one-to-one correspondence, but with the difference from Figure 3.11(a) that a session is never active when all outstanding data transfers are completed.

Since "call" is a lower level concept than "session" and since the way in which sessions are related to calls may be application dependent, it appears the session level must be application-dependent in general if the lower levels are to be hidden by the session level.

Figure 3.12 illustrates this point. The session level may be different for different applications. And only applications which are closely related share the same session level. Or the session level may be parameterized to provide different types of session levels for different applications. Or a standard session level makes the services of the lower network control level explicitly visible to the application level. In the latter case, the purpose of the session level is unclear, since its functions are effectively performed by the application layer.

The view that different types of sessions may be conducted for different applications is consistent with ISO, which provides for session context negotiation.

Although the session level of the first TRADEX prototype is too simple for general open system interconnection, not all of the functions suggested in Section 3.4.1 are necessarily appropriate. Some functions are better left to higher levels. In particular, flow control, checkpointing and recovery of files and messages originating at the highest application level are handled more appropriately by levels above the session level. The session level cannot distinguish messages containing application level data elements from control messages originating in the presentation and lower application levels, whereas this distinction is desirable for flow control, checkpointing and data recovery.

3.4.3 Sessions in Electronic Mail

In the MSDL's experimental electronic mail application, few issues arise at the session level. Only X.25 is used. And sessions are not visible to the operator. Accordingly, issues with respect to application and network dependence do not arise. Separate sessions are established for sending and receiving mail. The mail sessions are not dialogues at the application level; mail flows only one way in each session. However, two-way communications exists during a session for control purposes.

In the experimental system, sessions are periodically established for the specific purpose of sending or receiving batches of mail; one call is established per session and lasts for the duration of the session. In the general case send and receive sessions may be simultaneously established and (perhaps) may share the same call.

The services required for the experimental system are the basic ones of Figure 3.7. And the message flows are the corresponding basic ones of Figure 3.8. The fields required are those of Figure 3.9, with billing, password, calling terminal ID and session status fields omitted.

Figure 3.13 shows the FSM which controls the functions of the mail session level.

Note that the session level does not perform flow control, checkpointing or recovery of higher level data units but only clears the session if abnormal operation is detected.

For the experimental mail system, the approaches towards handling session failures were analyzed and it was determined that the session level would need excessive quantities of storage if it were to control session reset; therefore, data recovery responsibility was left to the higher levels, which have direct access to the data and have the facilities for determining the information lost during the failure.

Thus session failures are handled as follows:

1. Each session message contains two sequence indicators: M(S), M(R). The M(S) value received must preserve sequentiality with the previously received message. This check should never fail as sequencing is assured by the X.25 transfer level used. Any discrepancy would likely be due to a critical failure on either the hardware or software of the mail station itself. If a sequencing error does occur then both ends must attempt to adjust and restart at a common point to prevent loss or duplication of messages. As it is impractical for the session level to keep a window of messages in its environment, this restart cannot be executed without presentation level intervention. Therefore the session level will abort the session, clear the channel and notify the higher level through returning status values of the session level service requests.

2. Timeouts may occur in communicating with the remote presentation level. If the interval is made sufficiently large then it may be assumed that some failure has occurred and timeout retransmissions need not be attempted. A timeout on a command-response pair such as session request-session confirmation would leave each end in mutually unknown states from which the least complex recovery would be obtained by aborting the session.

3. Timeouts may occur due to lower level failure. Again the solution adopted is to abort the session whenever a session command-response pair times out. This approach eliminates the possibility of recovering a session after link failure; however, it guarantees the state of a session will never be ambiguous on an end-to-end basis.

3.4.4 Sessions in Communicating Word Processors

The main issues particular to communicating word processors arise at the higher levels. Accordingly we defer discussion to Sections 3.5, 3.6 and 3.7.

3.5 Presentation Level

3.5.1 Introduction

The main purpose of the presentation level is the delivery of application level data units in a form understandable by the application and at a time when they are expected by the application. How to decide on allocation of particular data transfer control and data formatting functions between the application and presentation levels is in many cases unclear. Particular approaches are presented here and in Section 3.6 and the broader issues are discussed in Section 3.7.

The presentation levels in the TRADEX and Electronic Mail Systems are first discussed in Section 3.5.2 and 3.5.3. Then Section 3.5.4 briefly discusses some issues specific to communicating Word Processors.

3.5.2 Presentation Level in the TRADEX Environment

In the initial TRADEX environment, which involves like terminals talking to each other, the data encoding is the same for all terminals and is commonly understood by all terminals. There are no encryption or data translation requirements of any kind. And it was decided by the design team that file transfer control is an application-level function. Accordingly, in the initial TRADEX system, the presentation level is entirely missing.

Future TRADEX systems may require data translation between COSTPRO aligned forms and other trade data formats. However, this translation would take place most appropriately at the application level, because it involves semantic interpretation of application level data elements. This subject is discussed further in Section 3.6 and 3.7.

3.5.3 Presentation Level of an Experimental Electronic Mail System

A design goal for the experimental system was to separate very clearly the application and presentation layers as follows:

- The application layer should be concerned with delivery and reception of pages of mail documents to and from the "mailbox". The nature and contents of each page are important to this level, but not the transfer control mechanisms.

- The presentation layer should be concerned with all details of control of transfer of the pages of each piece of mail and of the end-to-end acknowledgement of the entire mail document. The nature and contents of each page should not be of concern to this level.

However a requirement which compromises this goal was to provide the capability to refuse incorrectly addressed mail before the contents have been transmitted. This requires a separation of envelope page and content transfer services and functions in the presentation layer to allow the application layer at the reception end to check the envelope page and refuse the mail. The result is a need for the presentation layer to differentiate between envelope and contents pages. Rather than requiring the presentation layer to examine the pages, separate services are defined to send and receive envelope and content pages. The presentation layer performs different functions for these different services.

External services of the presentation level of an experimental electronic mail system which satisfies this latter requirement are illustrated in Figure 3.14 and explained below:

On Transmission

- Request session (and wait for establishment)
- Request credit for a document of n pages (and wait for credit)
- Transmit envelope page of document (and wait for acceptance)
- Transmit remainder of document (and wait for successful checkpointing)

On Reception

- Wait for a session
- Wait for an envelope page
- Send an envelope page acceptance
- Wait for remainder of document

Internal functions of this layer are as follows:

- Service application level requests

- Checkpoint entire documents by means of checkpoint request and checkpoint response messages at the end of each document transmission and reception cycle, respectively. In the experimental system, the checkpoint request message is used to solicit acknowledgement from the receiver that the mail document has been received and stored on disk. Failures result in retransmission of the entire document. Although not implemented in the experimental system, the message could also be used to recover from low level session failures by partial retransmission.

These functions are implemented by messages as shown in Figure 3.15. The FSMs controlling the performance of these functions are shown in Figure 3.16 and 3.17.

Note the somewhat awkward split between the application and the presentation level. Contrary to the goals stated above, there is a division of the transfer control responsibility between the application and presentation levels. The application level must now itself perform the sequencing associated with envelope pages. In this light, it is questionable whether this so-called presentation level is not actually the lowest application sub-level. This would be consistent with the approach described in Section 3.5.2 for the TRADEX terminal.

This simple, experimental electronic mail system does not embody all features required of the presentation level of a general system. Desirable features and components of a general system are illustrated in Figure 3.18 and discussed below:

1. The presentation level should include transfer control of documents on a page-by-page basis. This is illustrated by the example system described above.

2. The transfer control portion of the presentation level should include checkpointing and flow control of application-level data elements. Checkpointing (in simple form) is included in the experimental system described above. Explicit presentation-level flow control is not included in the experimental system, except for the credit grant mechanism, which provides high level flow control on a per-document basis. Reliance could be placed on low-level flow control mechanisms (in the network and transport levels), but these only provide bulk flow control over all concurrent sessions using a particular communications channel or link; they do not provide the possibility of flow-controlling individual concurrent sessions separately. Such individual flow control could be very important if facsimile services were combined with electronic mail in a single terminal. Facsimile services are capable of delivering very large amounts of data without advance warning as to length. The issues associated with facsimile transmission and its possible integration into multipurpose terminals are deferred until Chapter 5 and Chapter 7.

3. The transfer control portion of the presentation level could be shared by different applications. For example, if they were shared by electronic mail and file transfer applications, pages would correspond to records, documents to files and envelope pages to file descriptor messages. While desirable, this may not be possible if unforeseen application issues arise in the presentation layer, such as the special treatment of envelope pages in the simple electronic mail system. Careful study is required before agreeing or disagreeing with this item. Figure 3.19 illustrates the general nature of the alternatives.

4. <u>Presentation should include encryption control of page bodies</u>. Although no encryption control is present in any of the MSDL systems, it clearly belongs in this level.

5. <u>Presentation includes translation control</u>. This is primarily a detailed encoding/decoding function associated with the use of different character sets and character sequences for communications and applications. Translation control is discussed further in Section 3.5.4 in the context of Communicating Word Processor Systems and then again in Sections 3.6 and 3.7 in a broader context.

3.5.4 Communicating Word Processor Presentation Level Issues

As discussed in Section 3.5.3, the presentation level is a logical place for some translation control functions. Communicating word processors are characterized by major code translation problems in three areas (as discussed in detail in Ref. 3.2):

1. Incompatible text control codes and page format descriptions between different word processors;

2. Incompatible codes between internal text files and communications in individual word processors (usually because internal codes require 8 bits per character);

3. Unsuitable communications protocols which make assumptions about text formats (for example about line lengths).

There are two major issues with respect to solving these problems within the Open System Interconnection framework:

1. What translation functions are required?

2. In which level should these functions reside?

First consider code set translation problems. Can these problems be solved by Teletex-like mechanisms (Chapter 4), using the G0, G1 and G2 code sets?

The proposed G0/G2 set can handle all text requirements between word processors studied by the MSDL, provided it is augmented by additional information which specifies how printable characters outside G0/G2 set are represented within the set. It is possible to do this via a G1 set which is agreed upon between two parties and negotiated at session establishment.

Most text control characters could be handled in the same way using extensions to the C0 control set if a standard set of control characters could be defined for a set of standard word processing functions. However, some functions are represented by a string of control characters or by mixed strings of control and text characters. These are completely different between manufacturers.

Format information may be included in different ways within a text page (on individual lines or at the head of the page) or it may be outside the text proper (as a binary file header). There is at present no universally applicable way of representing format information.

What is needed is an international agreement on the representation of text control characters and formats. Some progress is being made in this direction [5].

One possible way of arriving at a standard would be to adopt a standard language for word processing functions. Such a language approach is used in the Unix word processing system [6] in which all text formatting and control commands are expressed in language form as part of the text itself. It is also used in the TEX word processing package developed by Knuth [7]. While hardly desirable as the input mechanism for the usual office users of word processors, this approach holds promise for communications.

The question of which level these translation functions should reside in is left for further discussion in Sections 3.6 and 3.7. With the introduction of language and semantic issues, the translation functions may be more appropriate at the application level.

3.6 Application Level

3.6.1 Introduction

The application level handles all remaining functions associated with particular applications. As we have seen, however, it is not always clear where the division should lie between presentation and application-level functions. Sections 3.6.2, 3.6.3 and 3.6.4 describe particular application-level approaches for TRADEX, Mail and Communicating Word Processor applications, respectively.

3.6.2 TRADEX Application Level

In the TRADEX application, operators make communication requests directly to the application level to transfer file/screen data to remote terminals. Thus the application level may be said to provide services (in the form of keyboard commands) to the operator, as illustrated in Figure 3.20.

Application level message flow is shown in Figure 3.21 and corresponding FSMs for control of application level functions are shown in Figure 3.22 and 3.23. Note that the application level here performs the equivalent of the envelope page and checkpoint functions of the presentation level for the experimental electronic mail system described in Section 3.5.3.

Missing from the basic system are format conversion services and functions between different trade data representations which may be required for international trade, such as CTDAS, EDI and TDI. CTDAS is the Canadian Trade Document Alignment System developed by COSTPRO [8]. EDI is a standard for formatting trade data for Electronic Data Interchange which is in limited use in the United States [9]. TDI is a Trade Data Interchange standard which has been adopted as a guideline for use in Europe. The main reason for this omission is that conversion requirements between these and other different representations are not at present well defined. COSTPRO is working on defining conversion tables between CTDAS and EDI. It appears that changes to EDI may be required and it is understood that COSTPRO is negotiating such changes. The effect of these changes will be to bring EDI closer to TDI. A preliminary joint specification has already been released [10]. COSTPRO believes that conversion between CTDAS and TDI will present few problems. Therefore conversion between CTDAS, modified-EDI and TDI should ultimately be possible.

When the conversion tables are defined, their use in the TRADEX environment will probably be limited to reformatting files in the application level before transmission, leaving all lower levels in the communication system untouched. This can be accomplished by including the type of file format in the file descriptor message and by arranging that a negative response to this message includes a field to specify the type of format acceptable to the remote terminal.

The issues associated with the allocation of functions to the application or presentation layer have been discussed in Section 3.5 and are discussed further in a broader context in Section 3.7.

3.6.3 Electronic Mail Application Level

In electronic mail, the application level of the communications system is enabled periodically by the central operating system. No other external services are provided, as shown by Figure 3.24. Once activated, the SEND and RECEIVE side of the application level functions without operator intervention. They use the services of the presentation level to send and receive envelope pages and data pages as shown in Figure 3.25. The FSMs controlling the internal functions of the application level are shown in Figure 3.26 and 3.27.

The issues associated with the allocation of functions to the application or presentation layer have been discussed in Section 3.5 and are discussed further in a broader context in Section 3.7.

3.6.4 Application Level Issues for Communicating Word Processors

The main issue is how to allocate translation control functions between the application and presentation levels, as discussed briefly in Section 3.5. This issue is discussed further in a broader context in Section 3.7.

3.7 Summary of Issues Arising from MSDL Experience

3.7.1 Introduction

The foregoing sections of this chapter have raised a number of issues with respect to the levels of the Open System Interconnection model, in the context of MSDL experience with particular applications.

These issues are now summarized and discussed with particular reference to the implications for multi- purpose terminals. As before, issues are discussed level-by- level from the bottom up, starting with the Transport level in Section 3.7.2 and proceeding through the Session, Presentation and Application levels in Sections 3.7.3, and 3.7.4.

3.7.2 Transport Level Issues

The main issue with respect to the Transport level appears to be: Is it necessary? In MSDL systems it has been used to provide a uniform procedural interface to a variety of networks, with no autonomous functions of its own. Deficiencies at the network level (as with X.21-like networks) which required autonomous activity to resolve were handled by the Session level. And end-to-end acknowledgements and timeouts of messages passed to the Transport level were handled by higher levels.

The authors' view is that such a standard, procedural interface is necessary, but that the question of whether it should reside at the Transport level, or not, is open. It could be provided as a standard interface to the Network level.

3.7.3 Session Level Issues

The main issues at the Session level arising from MSDL experience are:

- support of multiple concurrent sessions, possibly multiplexed over a single communications channel;
- degree of session recovery required from failure;
- flow control requirements;
- role of the Session level in negotiating capabilities for use by higher levels;
- necessity for different types of sessions.

We here comment on each of these issues in turn.

a) Multiple Concurrent Sessions

Support of multiple concurrent sessions appears to be necessary in general if multiple applications, or concurrent communication activities within a single application, are to be supported. A single-session-at-a time organization does not provide sufficient concurrency. A severe restriction of such an organization is the inability to establish a session for reception while another communication session is in progress.

Of course, multiple concurrent sessions in a terminal can be handled by multiple, distinct sets of session-level services, one set for each communications channel (virtual or physical) supported by the network level. Then each set of services forms a separate Session level which does not by itself support concurrent sessions. However, such an organization presents an awkward interface to higher levels. And it is not sufficiently modular; changes at the network level may affect the organization of the Presentation level.

The best solution appears to be the support by a single Session level interface (i.e., by a single set of Session level services) of multiple concurrent sessions, each with its own separate state information maintained internally by the Session level. Requests for Session level services must then reference a session identifier supplied by the Session level upon session establishment.

b) Session Recovery

It appears to be both inappropriate and unnecessarily complex to provide automatic recovery at the Session level of data belonging to higher levels when a Session failure occurs. It is inappropriate because in general only higher levels are specifically aware of the state of their own data. And it is unnecessarily complex because Session failures should occur relatively rarely and sufficient automatic recovery mechanisms can be provided by other levels to handle rare failures.

It should not be necessary to point out to the reader that unnecessary complexity may make it very expensive to implement layered protocols in microprocessor-based intelligent terminals.

It appears to the authors to be probably sufficient to return the session to a single known state, namely the aborted state, on an end-to-end basis, when session failure occurs and to leave data recovery to higher levels.

Session failure can be detected by sequence count violations and time-outs.

A user of a service will of course not wish to pay for data lost in an aborted session. This may be handled by ensuring that higher levels do not post charges through the session level billing service (see item (f)) until a higher level handshake has been completed.

c) Flow Control Requirements

Flow control on a bulk basis is handled by the Network Level and below. It will also be handled as required on an application basis at higher levels. There is no reason for including it in the Session level, in agreement with OSI philosophy.

d) Role of the Session Level in Negotiating Higher Level Capabilities

The Session level may have a role in negotiating translation and encryption control services for the Presentation level. However its role is unclear, because the nature of these services is still unclear.

The Session level may also have a role in negotiating the type of transfer control protocol to be used by higher levels. For example, a one-way mail session might provide different services to send higher level commands and responses so that it can identify session failure when a command is sent in the wrong direction. A two-way, free-running dialogue session, on the other hand, would not need to distinguish commands from responses and so would only provide a single service to send data.

e) Necessity for Different Types of Sessions

Mentioned above was the possibility of negotiating different types of session services for use by higher levels, for example to handle one-way or free-running dialogue sessions.

It may be necessary also to negotiate different types of internal session functions for the same service. This requirement was identified for the TRADEX example, but is more generally applicable. For example, in one type of session, establishing a session would result in placing a call at the network level which would not be cleared until the end of the session. In another type, a session could be automatically suspended (temporarily) by the Session level by clearing the call during periods of session inactivity.

f) <u>Other Issues</u>

Billing for higher level service (in cases where this is required) should probably be accumulated on a session basis by the Session level through a set of separate billing services provided to higher levels.

The same approach probably applies to the gathering of session statistics.

Password validation and sender authentications are logical services of the session level.

3.7.4 Presentation and Application Level Issues

Presentation and Application level issues relate to at least two areas:

- Transfer Control;
- Translation Control;

Translation control is first discussed below, followed by Transfer Control.

3.7.4.1 Translation Control: Two Approaches to Presentation/ Application Level Separation

How to separate the Presentation and Application levels with respect to translation control is in many cases unclear. Two approaches are possible: The first may be termed "functional separation"; the second may be termed the "virtual application approach". The two approaches and the issues associated with them are described below.

(a) Functional Separation

In this approach, the Presentation level is restricted to communications-related translation and formatting functions. Three principles may be proposed to guide this separation.

Principle 1

Any formatting or translation function between data formats and codes which may be used as the primary application level data storage formats and codes in other nodes of the network should be performed within the application level.

For example, in a TRADEX network, either or both CTDAS and EDI formats and codes could be used for primary data storage at the application level in various nodes of the network. Therefore conversion from CTDAS to EDI formats and vice versa resides most appropriately within the application level. And plans at present are that it will reside there when it is implemented.

To give another example, in a network of communicating word processors, text files could be stored using special format information and control codes peculiar to individual manufacturers. Or they could be stored using standard ASCII with all formatting and control information embedded in the text as language statements. Some systems may even require use of the language statements for text input at the operator level. Therefore conversion between special formats/codes and language-oriented descriptions should occur within the Application level.

A companion principle states that:

Principle 2
Any formatting or translation function which may require understanding of Application-level data semantics should be performed at the Application level.

For example, translations between EDI, CTDAS and TDI formats could involve application data semantics, due to differences in the freedom of formatting certain fields. So could translation between special word processing codes/formats and language descriptions.

Another principle states that:

Principle 3
Any formatting or translation functions performed solely for communication purposes should be performed within the Presentation level.

Examples of such functions are data compression and data encryption, if they are performed solely for efficiency and security of communications.

These principles provide for relatively clean separation of the Presentation and Application levels, leading to modular systems in which changes to one level do not affect another.

They also provide for the early definition of a small number of widely useable, relatively simple Presentation levels for ranges of applications instead of a large number of different, relatively complex Presentation levels.

This approach may be contrary to the ISO and CCITT Open System Interconnection proposals for the presentation layer, although this is not absolutely clear. However it appears to be implicit in the CCITT Teletex proposals seen by the authors to date, judging by the absence of any translate/format functions in the proposed Presentation level (see Chapter 4).

This approach could lead to more rapid international agreement being reached on Open System Interconnection standards than the second approach described below.

(b) The Virtual Application Approach

An entirely different approach, the bare outlines of which are suggested in the ISO Open System Interconnection Report, may be termed the "virtual application approach". In this approach, a virtualized standard version of a single application is agreed upon. This application may be implemented in detail at the Application level quite differently in different terminals. A function of the Presentation level is, then, to provide translation and formatting services which convert local representations into the appropriate representations for the virtual application, and vice versa. Communications is in terms of the virtual application only, as arranged by the Presentation level.

For example, in a TRADEX network, the EDI standards could be used to define a virtual trade data service. The Presentation level would then translate CTDAS codes and formats into EDI codes and formats and vice versa. Terminals using EDI codes and formats at the Application level would not require the translation.

To give another example, a virtual word processor could be defined in terms of a standard text description language, if one can be agreed upon. Then the Presentation level in any particular terminal would be required to translate local codes and formats into the language statements of the virtual word processor and vice versa. Word processors using these language statements at the Application level would not require this translation.

This approach requires that the Presentation level be customized for particular terminals. Because of the possible wide variation in particular data codes and formats at the Application level, this customization could require not just installation of appropriate code translation tables but also of algorithms and data files to perform major reformatting functions based not only on data syntax but also on data semantics.

Both the TRADEX and Communicating Word Processor applications would likely require such algorithms and data files.

Given that these functions are so heavily application-dependent, it may be more logical to perform them within the Application level using the functional separation approach.

3.7.4.2 Transfer Control

The issues associated with transfer control relate to:

- checkpointing and error recovery;
- credit management and flow control
- separation of services for sending Application level data descriptors and data.

We treat each of these issues in turn, below.

Checkpointing and error recovery are clearly suitable functions of the Presentation level.

Credit management here implies obtaining permission from a remote node before sending it data. The basis for granting permission may be varied: room in the Mail file, lack of TRADEX file name collision, etc. Thus credit management can provide a form of high level flow control on a per-application basis. The question is whether lower level flow control is required as well in the Presentation level. For example, in a document-based system credit management would be relative to documents whereas flow control might be desirable on a per-application basis in a multipurpose Terminal relative to parts of a document. This might particularly be the case for facsimile. Further discussion of this issue is left until Chapters 5 and 7.

Separation of services in the Presentation level for sending data descriptors and data may be required if the descriptors are to be passed on for verification by the remote Application level before data is transmitted. In the TRADEX application this issue was resolved by doing it all at the Application level. In the experimental Electronic-mail System it was resolved somewhat awkwardly by providing separate presentation services and associating a different protocol with each service.

3.8 References for Chapter 3

1. "Canadian Trade Information Standards, Volume II, CTIS Devices - Functional Definition," COSTPRO, May 79.

2. R.J.A. Buhr, D.A. MacKinnon "Word Teleprocessing Interface," DOC Contract Report, March 1980.

3. "The X.25 Package Software Documentation," Edition 1.0, 1978-09-01, MSDL Technical Report, prepared for COSTPRO.

4. J.K. Cavers, "Implementation of X.25 on a Multiple Microprocessor System," International Communications Conference, Toronto, June 1978.

5. "Coded Character Set for Text Communication" ISO/TC97/SC 16N-1003 to 1008.

6. B.W. Kernighan, L.L. Cherry, "A System for Typesetting Mathematics," in the UNIX system documentation package.

7. D.E. Knuth, "Mathematical Typography," Bulletin of the American Mathematical Society, March 89.

8. "Canadian Trade Document Alignment System Manual," Second Edition, COSTPRO, 1978.

9. "The United States Electronic Data Interchange (EDI) Standards," Volumes I through V, December 5, 1978, Transport Data Coordinating Committee, U.S. Department of Transportation.

10. "The North American Electronic Data Interchange Standards, Volume V, General Communications Guide," COSTPRO, January 1980.

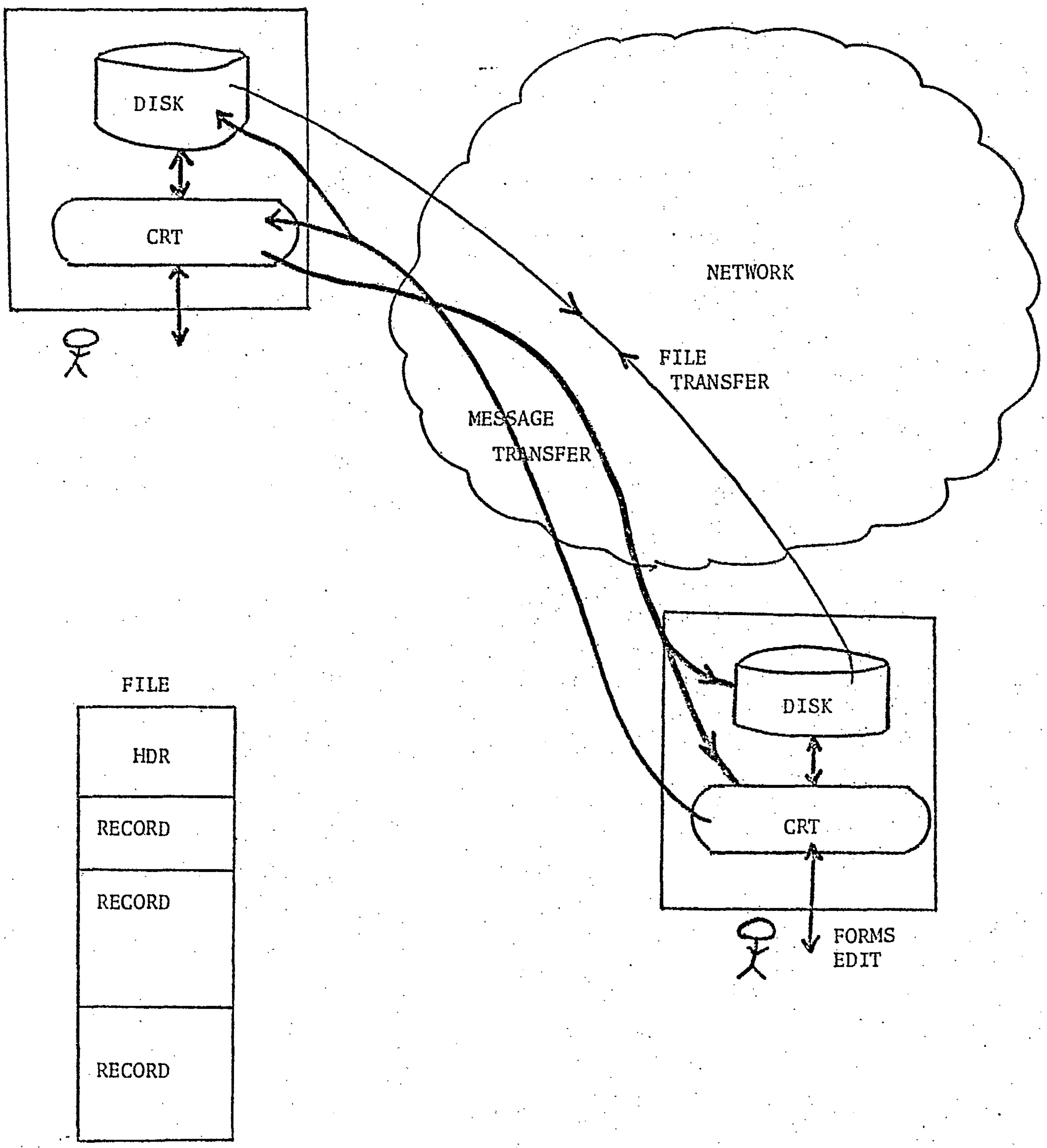


Figure 3.1: A TRADEX APPLICATION

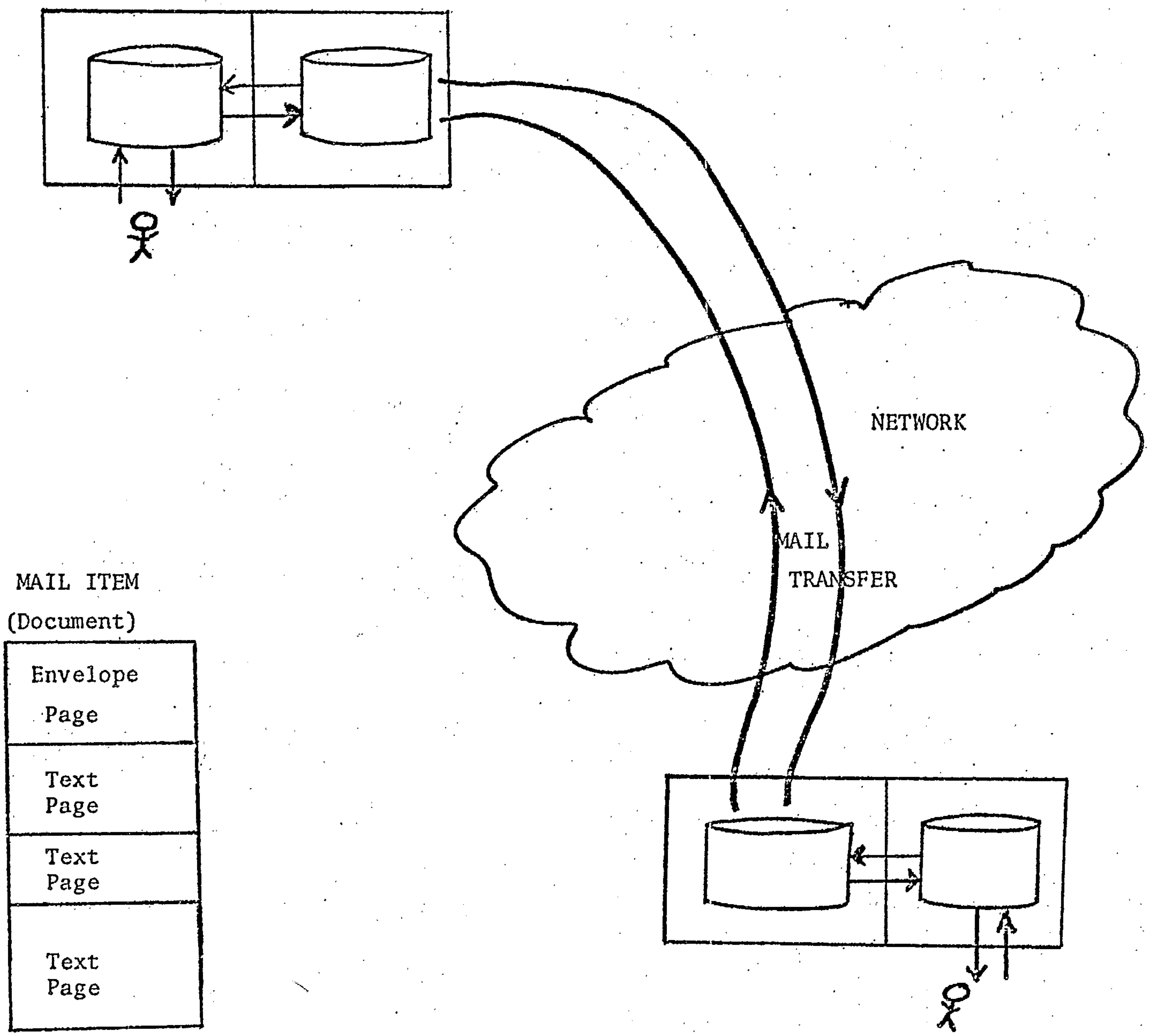


Figure 3.2: AN ELECTRONIC MAIL APPLICATION

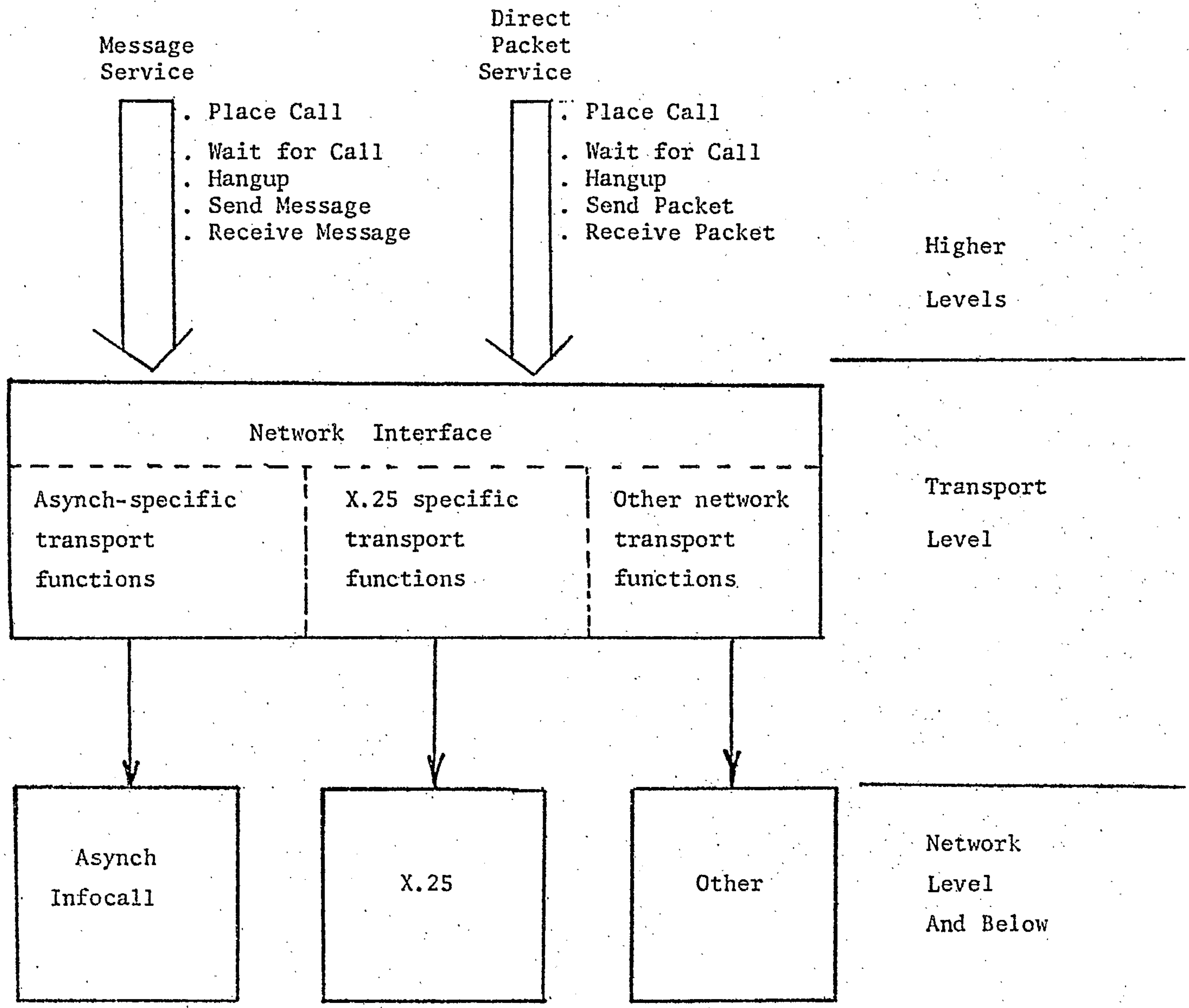


Figure 3.3: TRANSPORT LEVEL SERVICES

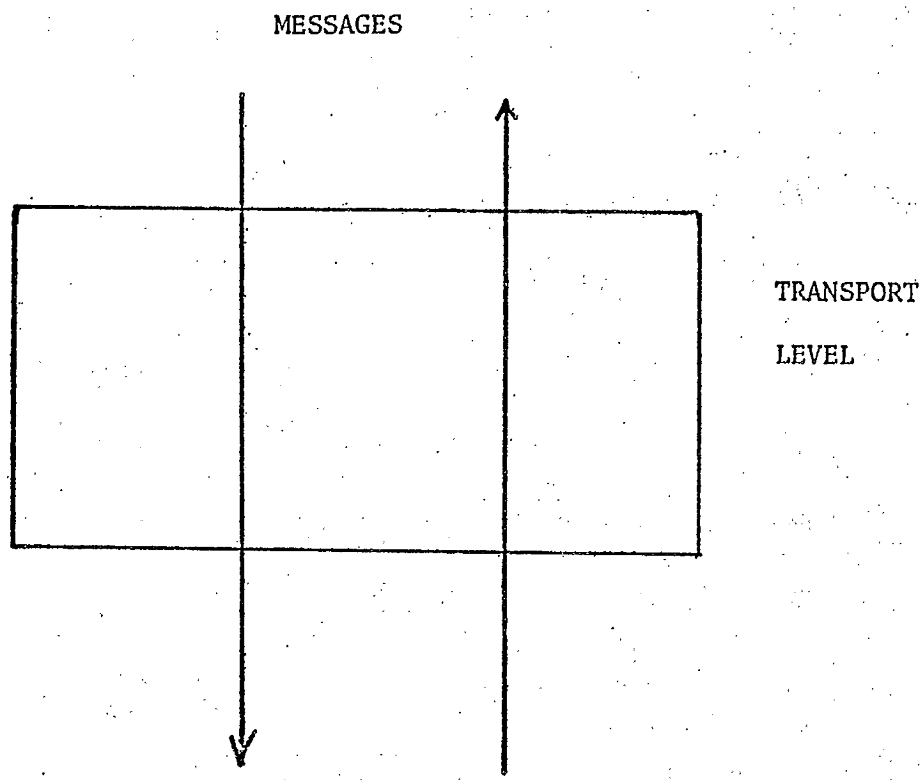


Figure 3.4: TRANSPORT LEVEL MESSAGE FLOW

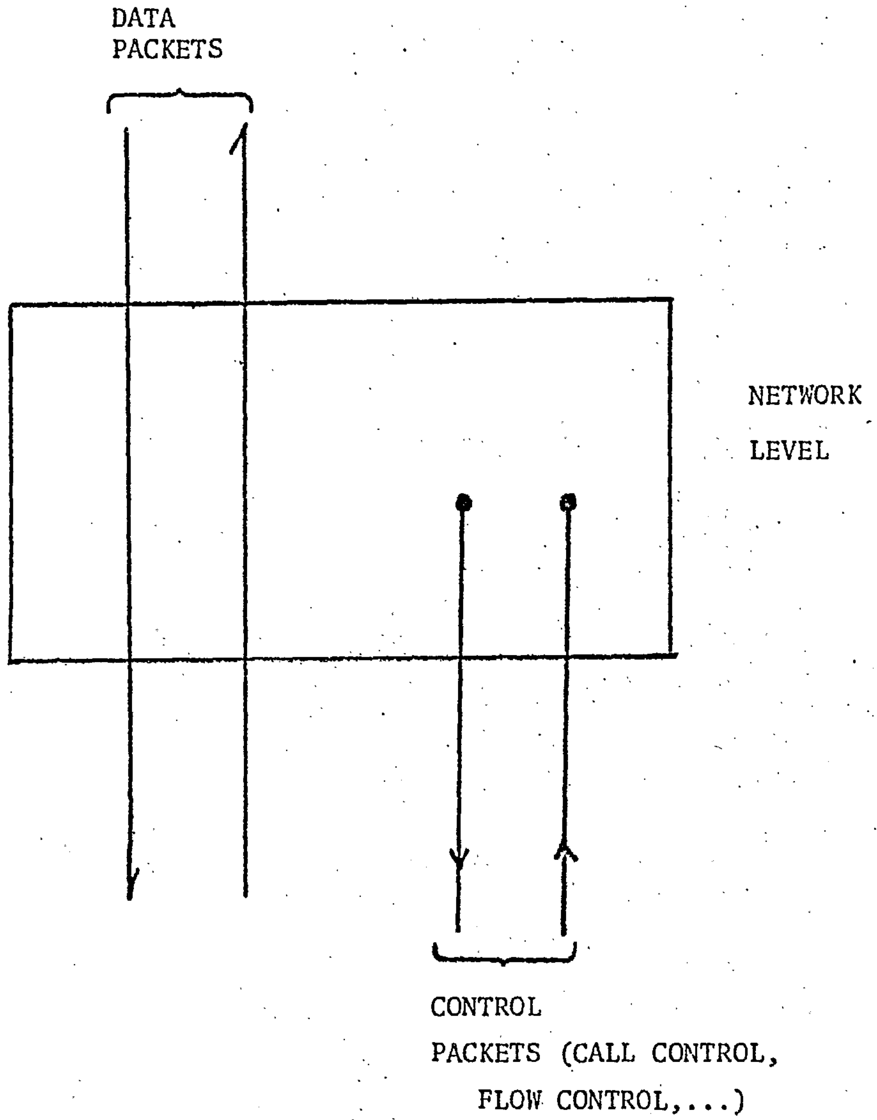


Figure 3.5: NETWORK LEVEL MESSAGE FLOW

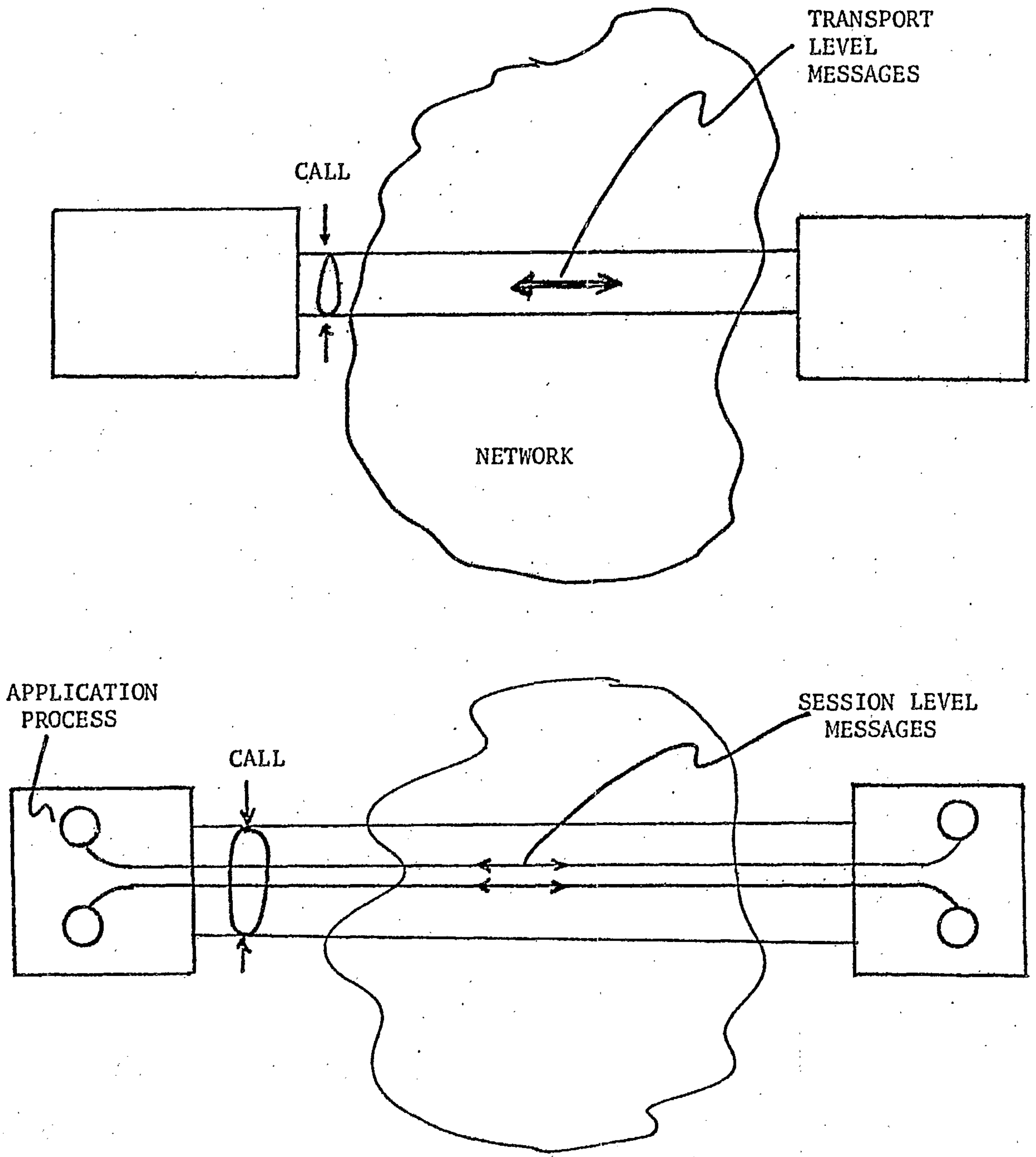


Figure 3.6: CALLS AND SESSIONS

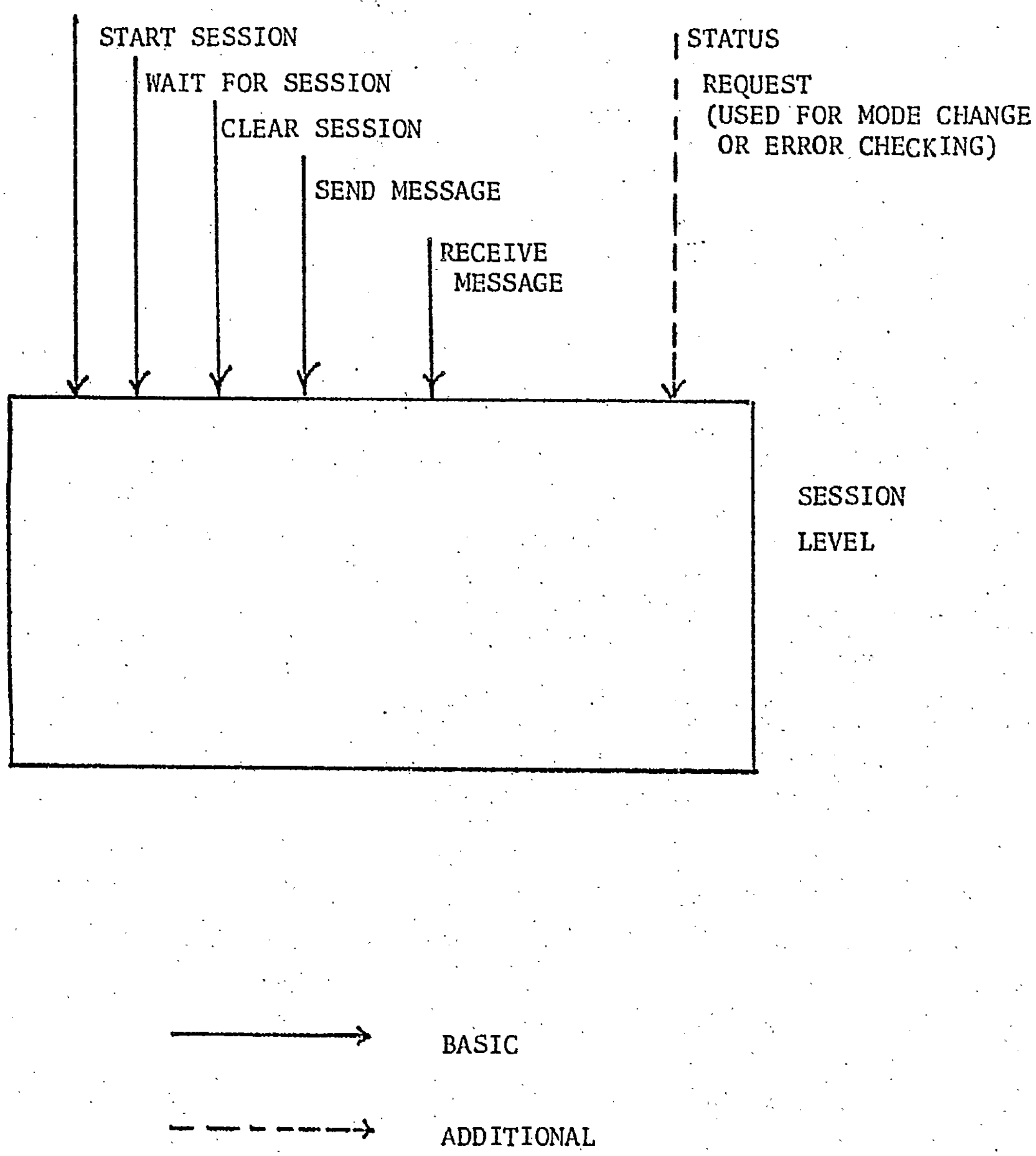


Figure 3.7: SESSION LEVEL SERVICES

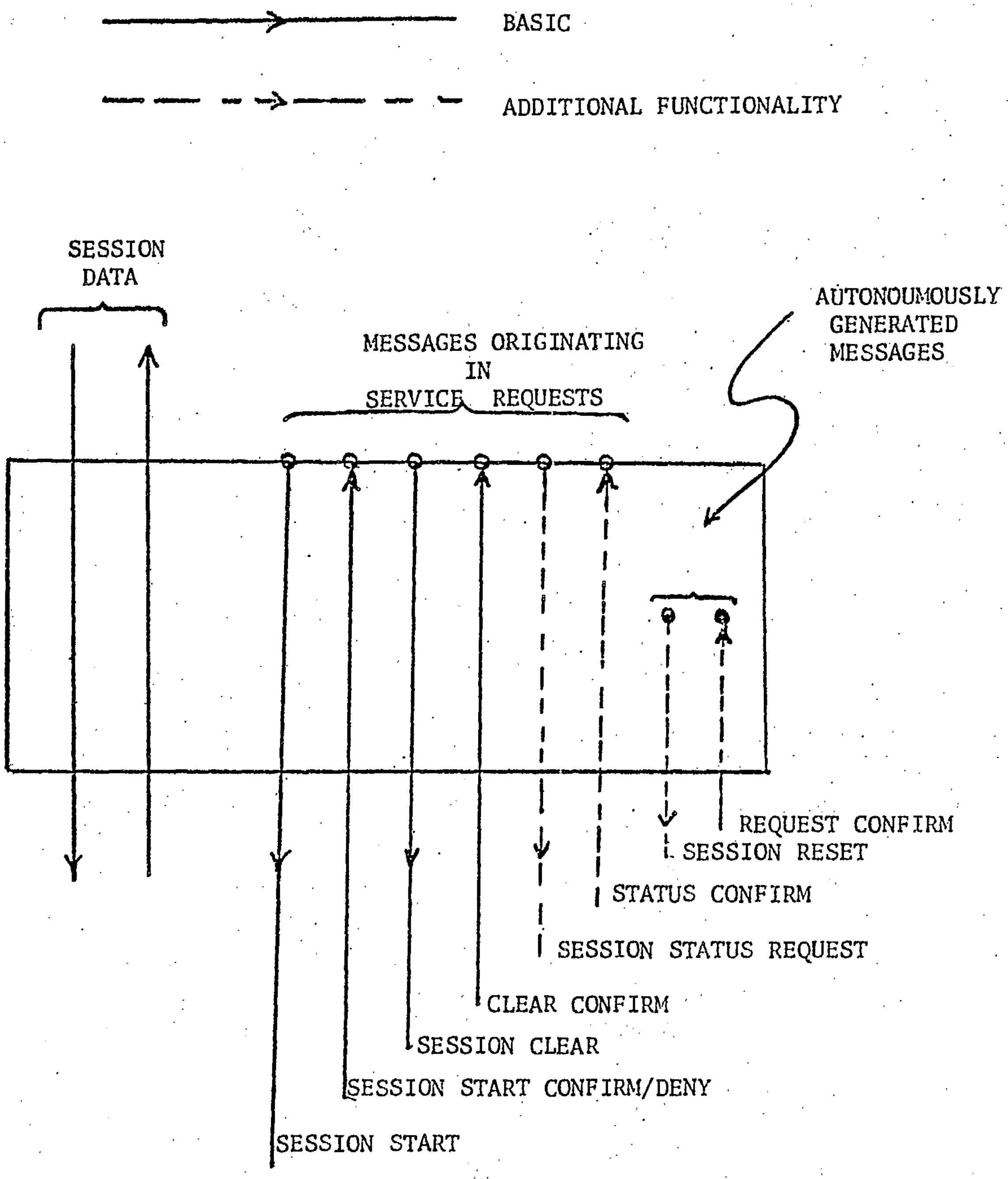


Figure 3.8: SESSION LEVEL MESSAGE FLOW

BASIC FIELDS

- SESSION ID
- SESSION TYPE (OR APPLICATION PROCESS ID)
- MESSAGE TYPE
- MESSAGE DATA
- CAUSE
- CALLING TERMINAL ID

OPTIONAL FIELDS

- BILLING
- PASSWORD

STATUS AND RECOVERY FIELDS

- SESSION STATUS
- MESSAGE SEQUENCE NUMBERS (SEND/RECEIVE)

Figure 3.9: CANDIDATE FIELDS FOR SESSION LEVEL MESSAGES

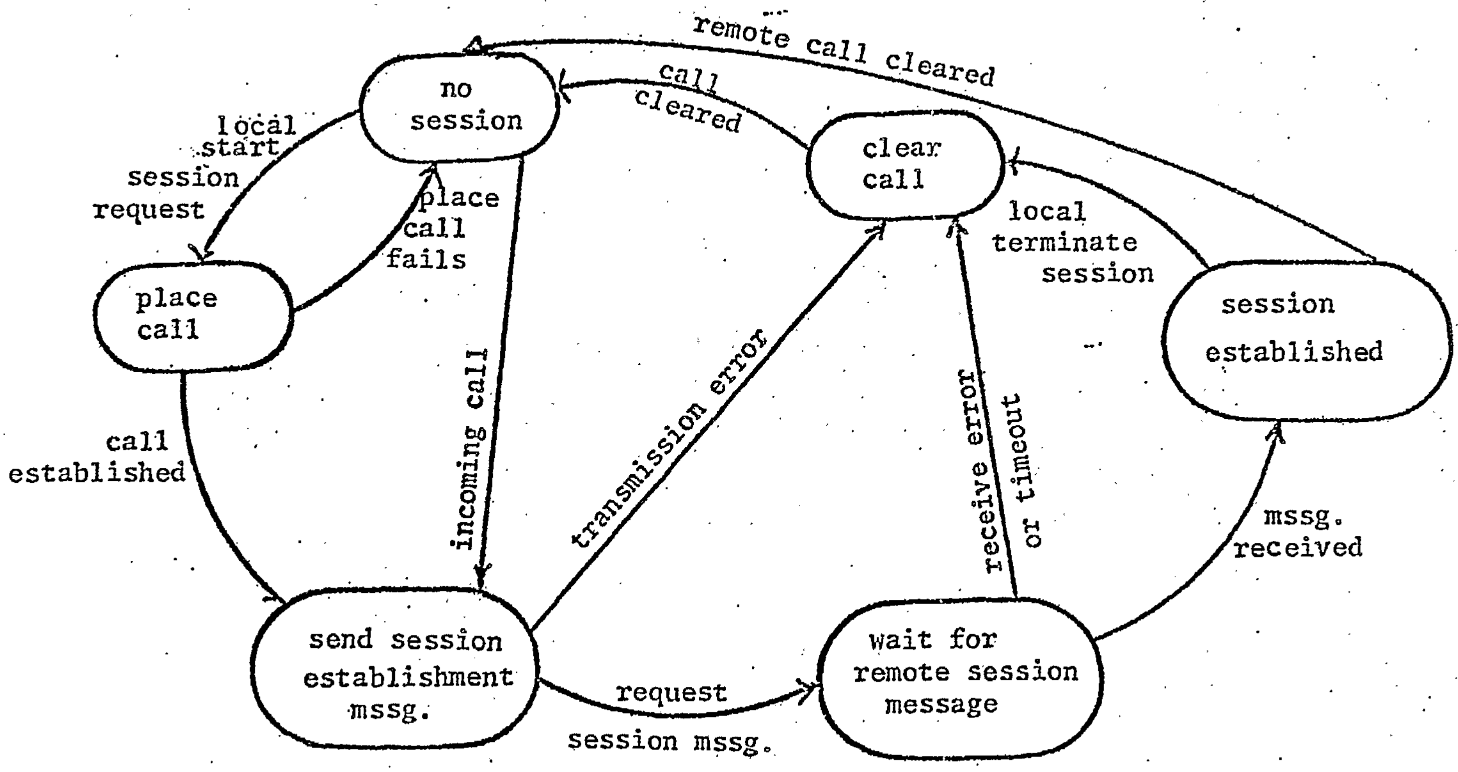
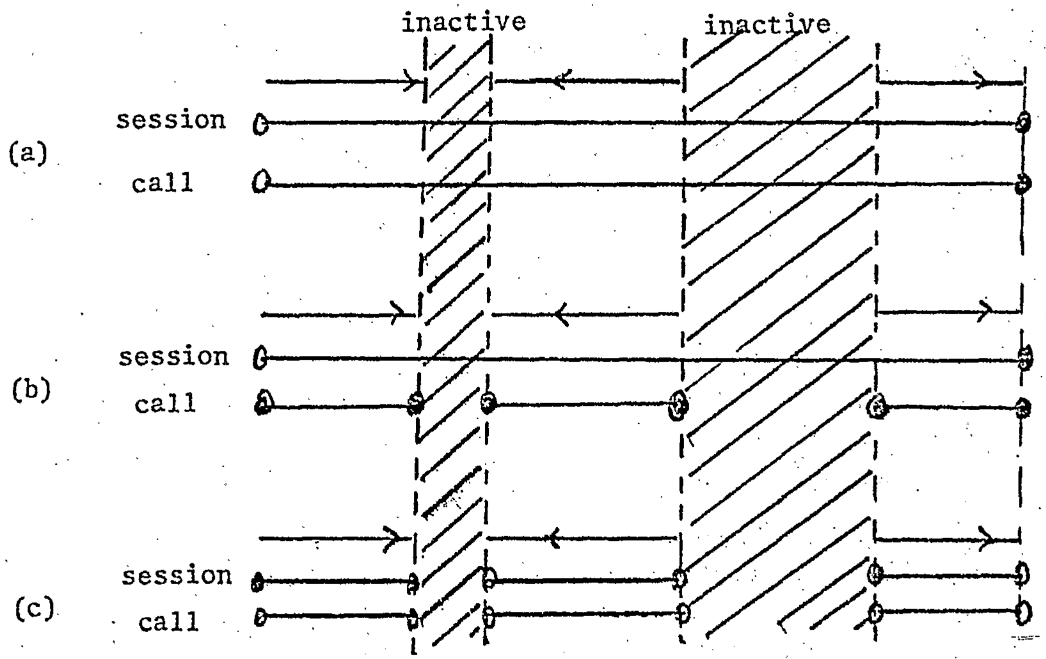


Figure 3.10: SESSION ESTABLISHMENT FOR X.21-LIKE NETWORK SERVICES



Note: ARROWS SHOW ALTERNATING DATA FLOWS IN A DIALOGUE

Figure 3.11: APPLICATION DEPENDENCE OF SESSION LAYER FUNCTIONS IN A TRADEX APPLICATION

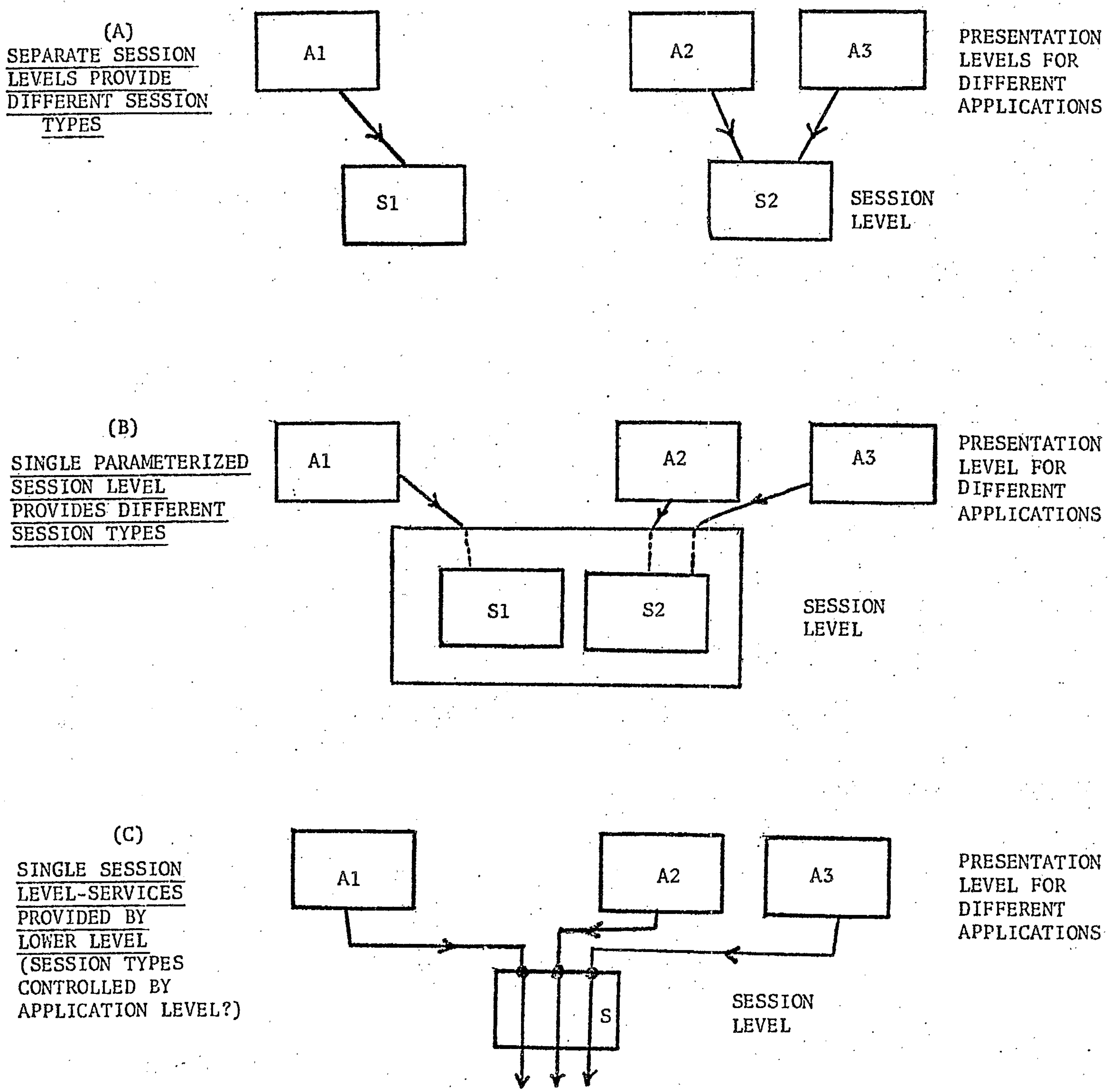


Figure 3.12: APPLICATION-DEPENDENT SESSION TYPES

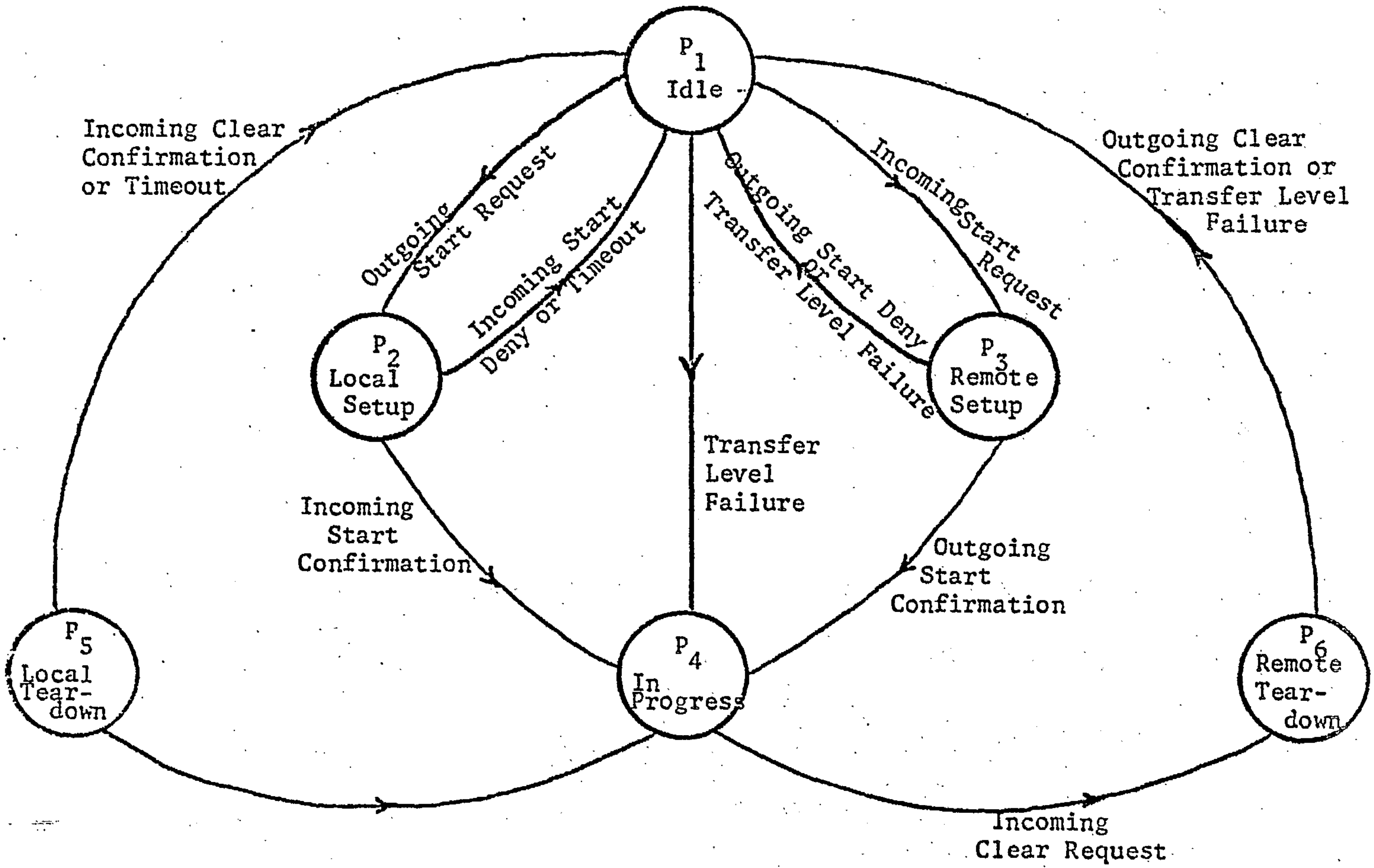


Figure 3.13: ELECTRONIC MAIL SESSION PROTOCOL

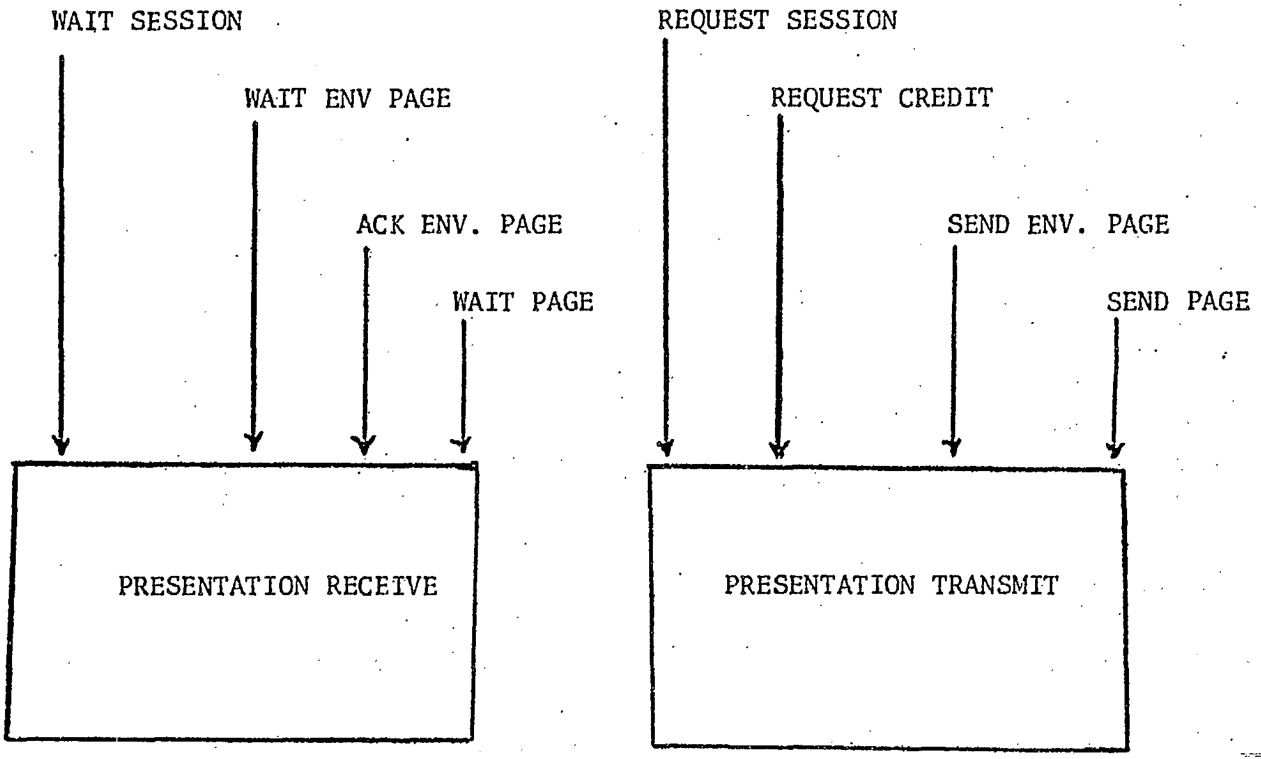
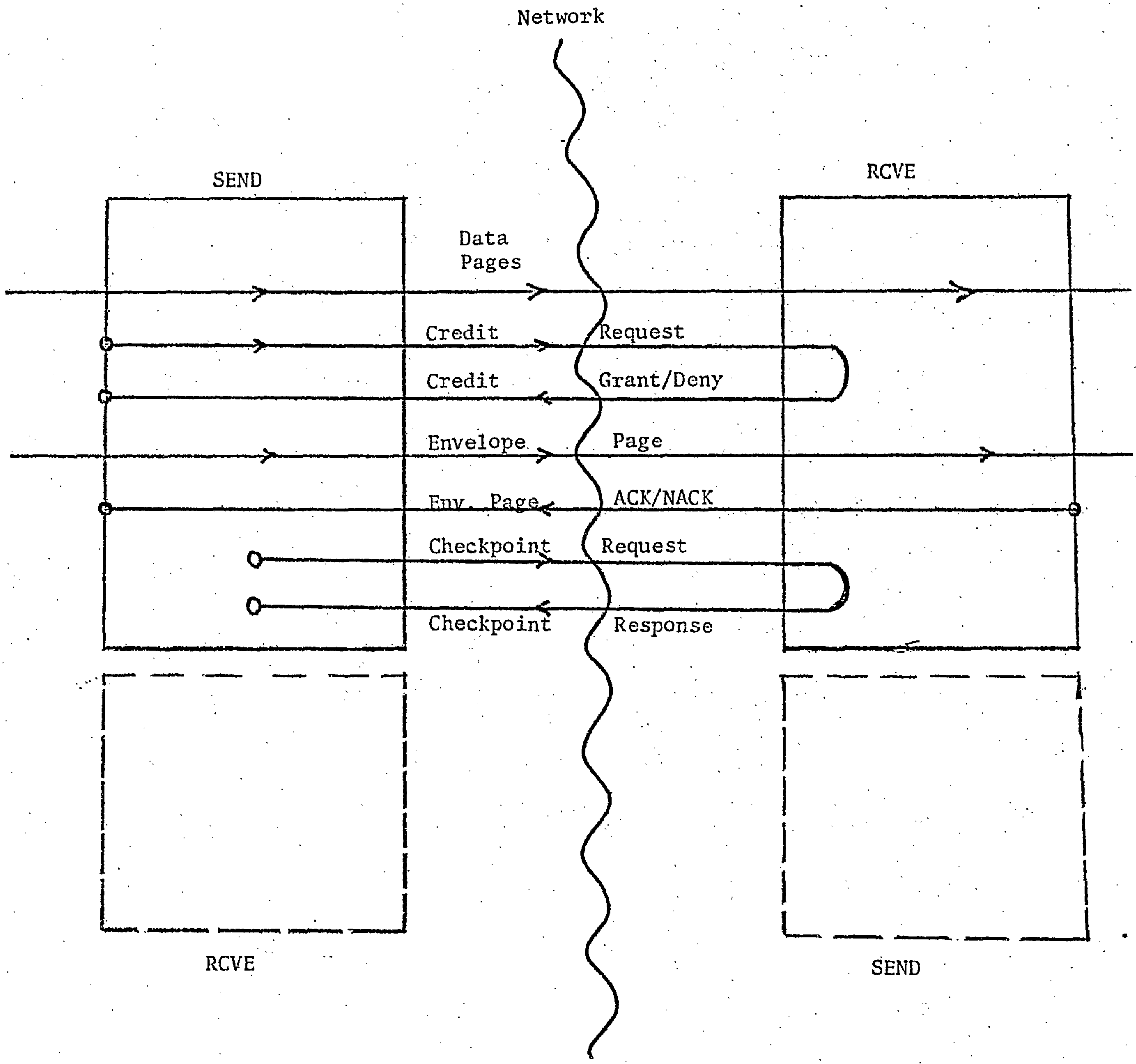
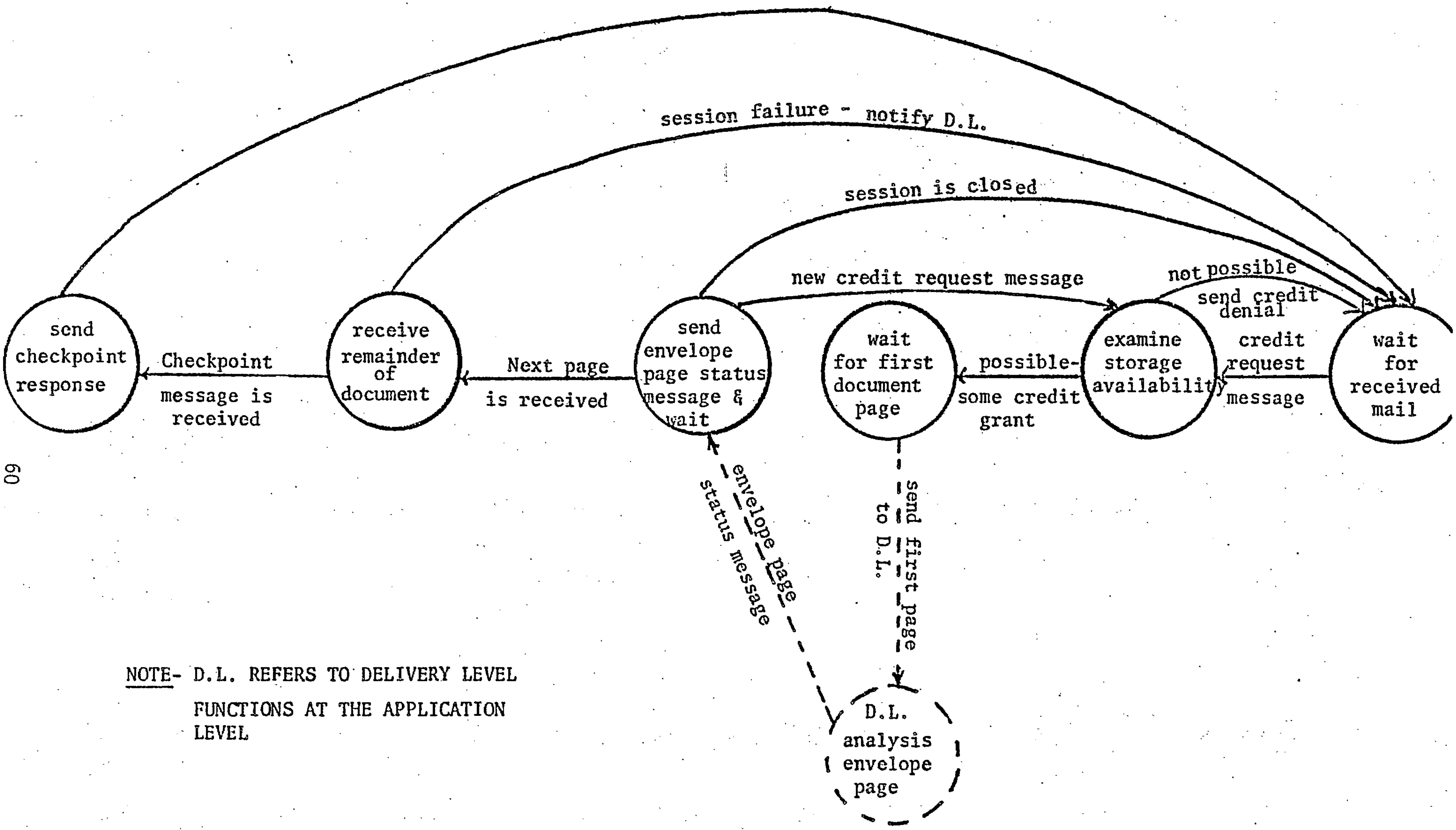


Figure 3.14: PRESENTATION LAYER SERVICES: SIMPLE ELECTRONIC MAIL SYSTEM



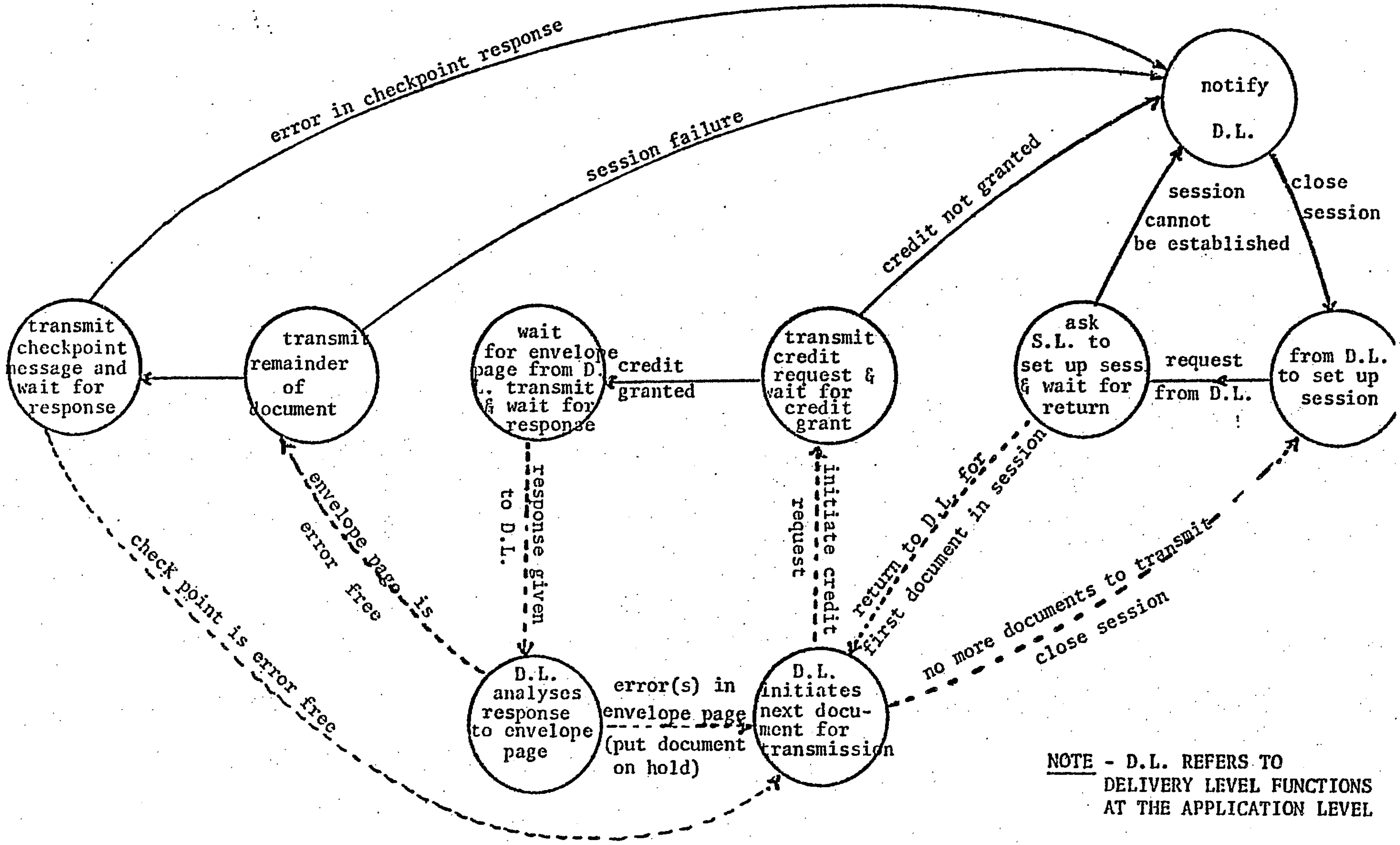
3.15: PRESENTATION LAYER MESSAGE FLOW:

SIMPLE ELECTRONIC MAIL SYSTEM



NOTE- D.L. REFERS TO DELIVERY LEVEL FUNCTIONS AT THE APPLICATION LEVEL

Figure 3.16: PRESENTATION RECEIVE FUNCTIONS



NOTE - D.L. REFERS TO DELIVERY LEVEL FUNCTIONS AT THE APPLICATION LEVEL

Figure 3.17: PRESENTATION TRANSMIT FUNCTIONS

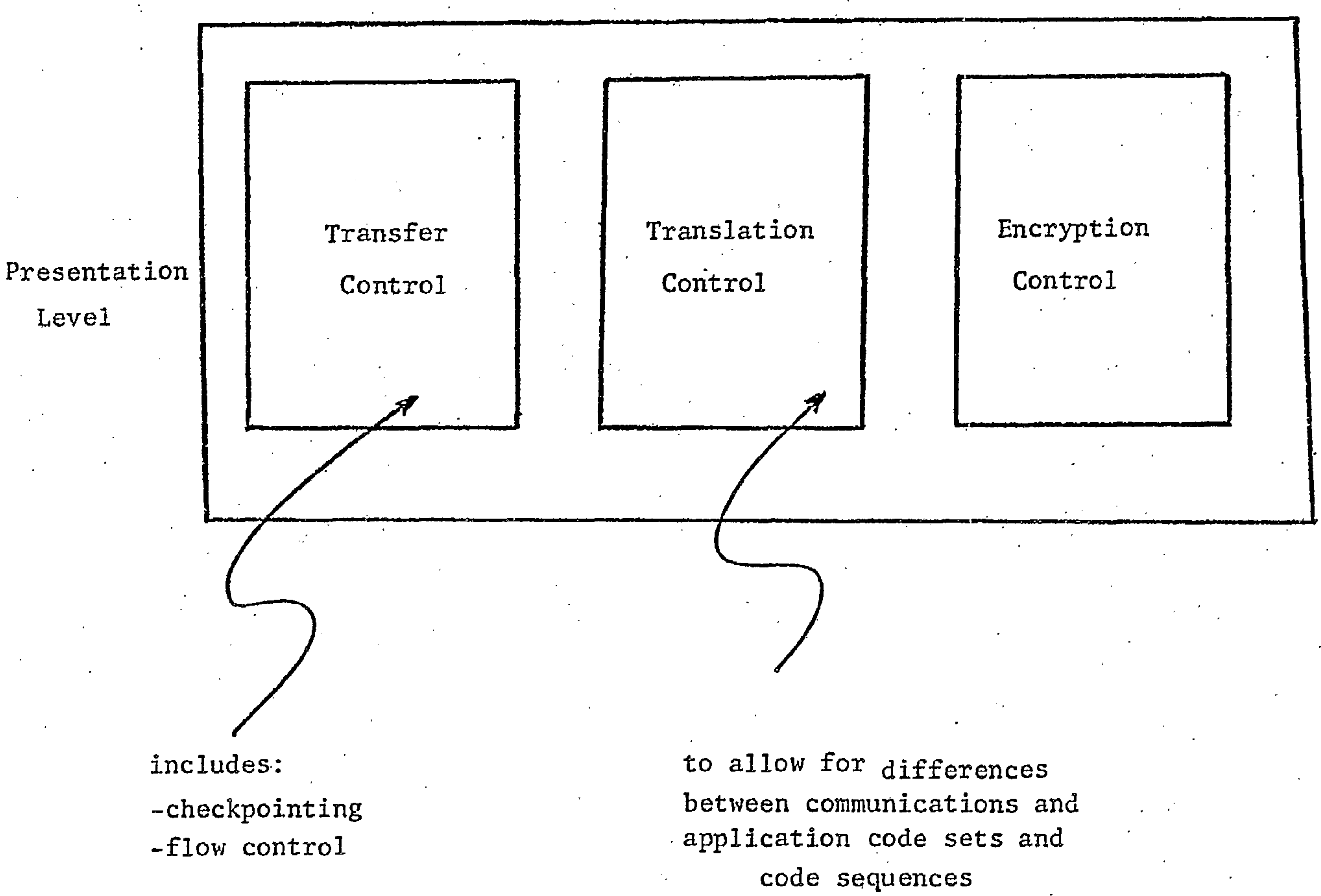


Figure 3.18: FEATURES OF A GENERAL PRESENTATION LEVEL: ELECTRONIC MAIL

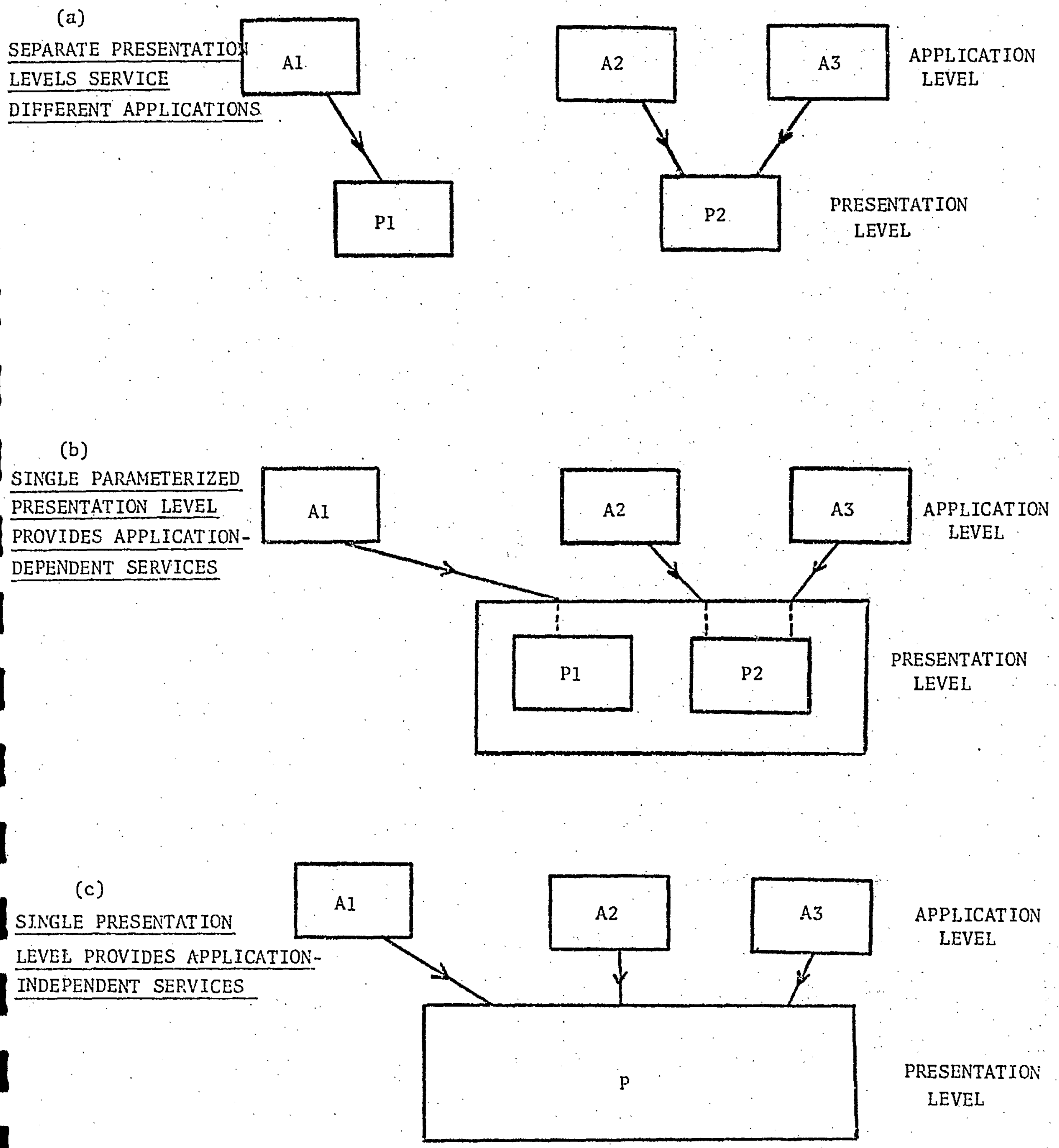


Figure 3.19: APPLICATION-DEPENDENT PRESENTATION LEVELS

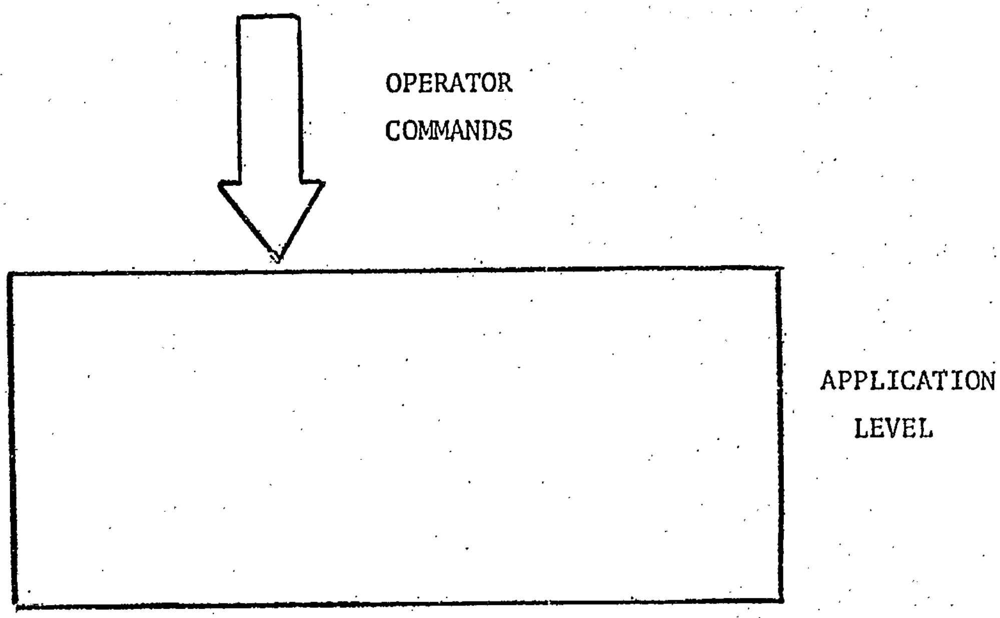


Figure 3.20: TRADEX APPLICATION LEVEL SERVICES

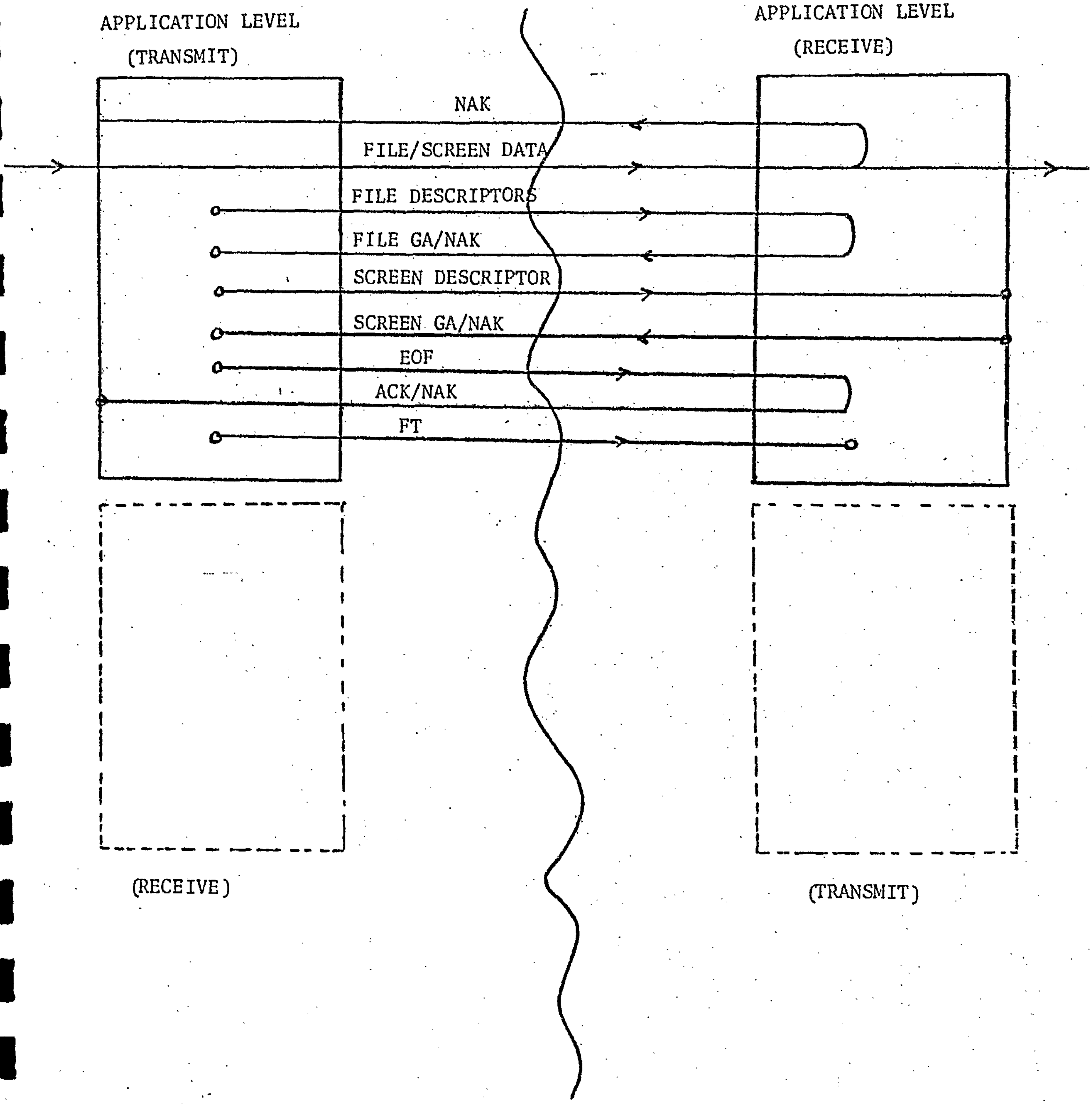


Figure 3.21: TRADEX APPLICATION LEVEL MESSAGE FLOW

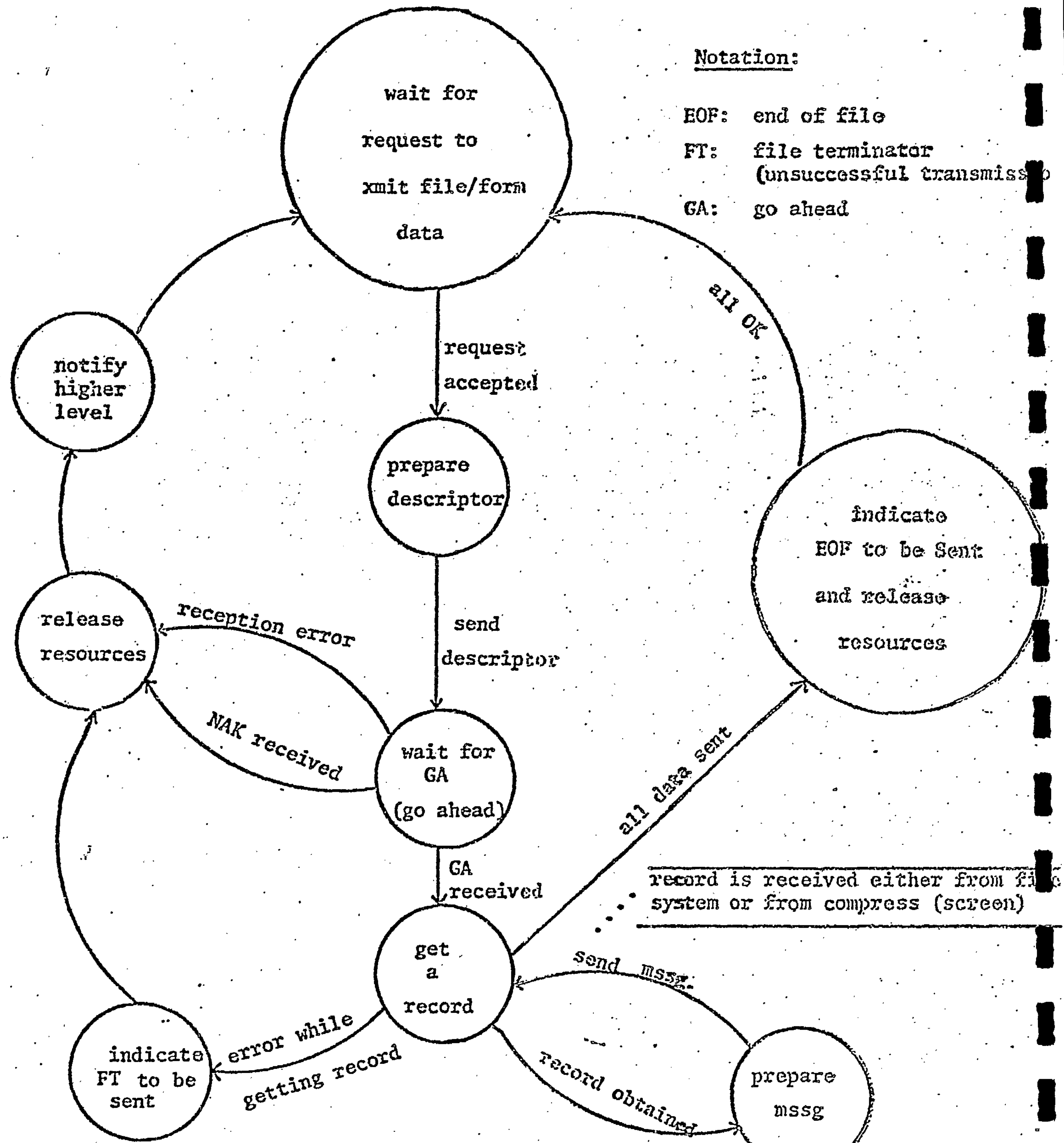


Figure 3.22: TRADEX APPLICATION LEVEL FUNCTIONS:
FSM FOR FILE/SCREEN DATA TRANSMISSION

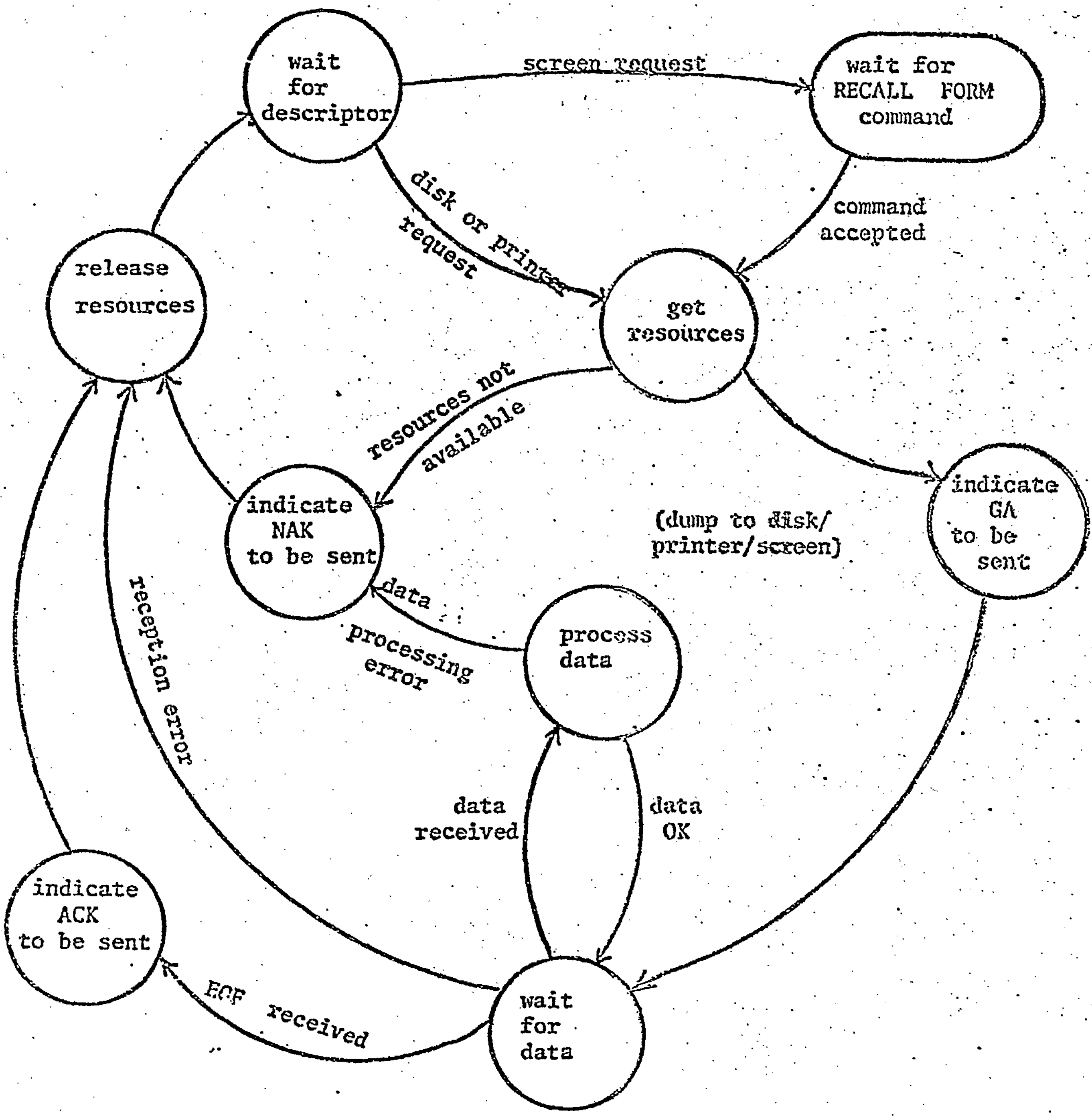


Figure 3.23: TRADEX APPLICATION LEVEL FUNCTIONS: FSM FOR FILE/SCREEN DATA RECEPTION

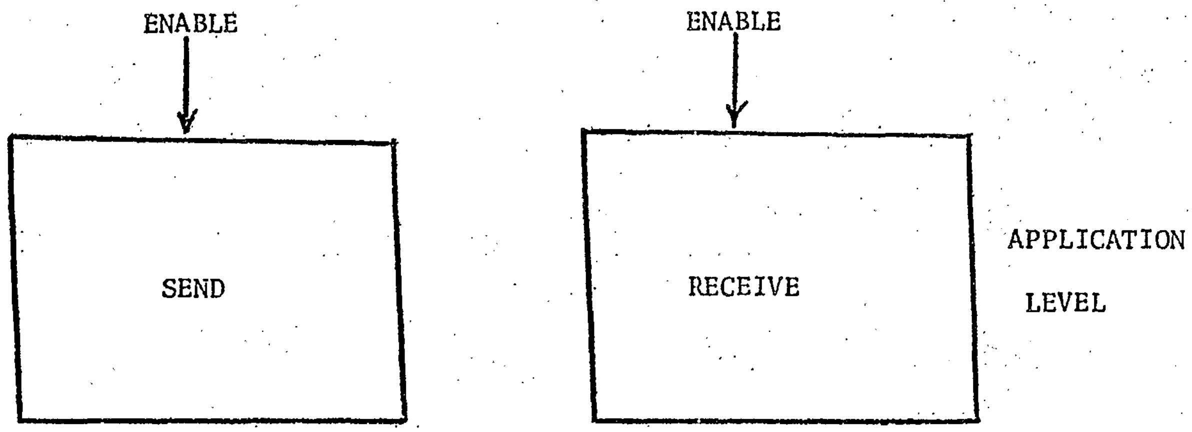


Figure 3.24: ELECTRONIC MAIL APPLICATION LEVEL SERVICES

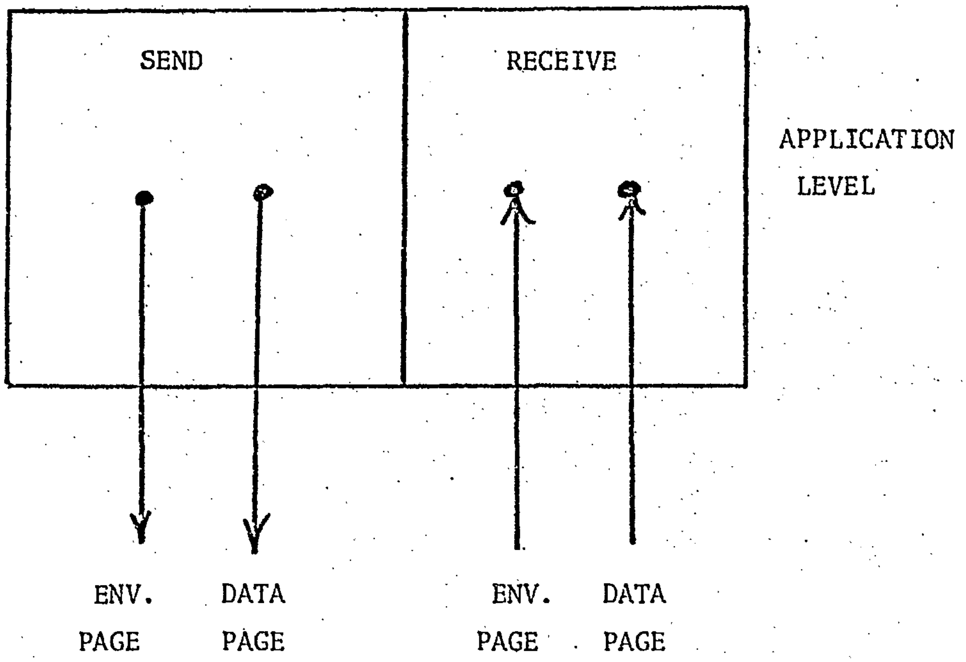
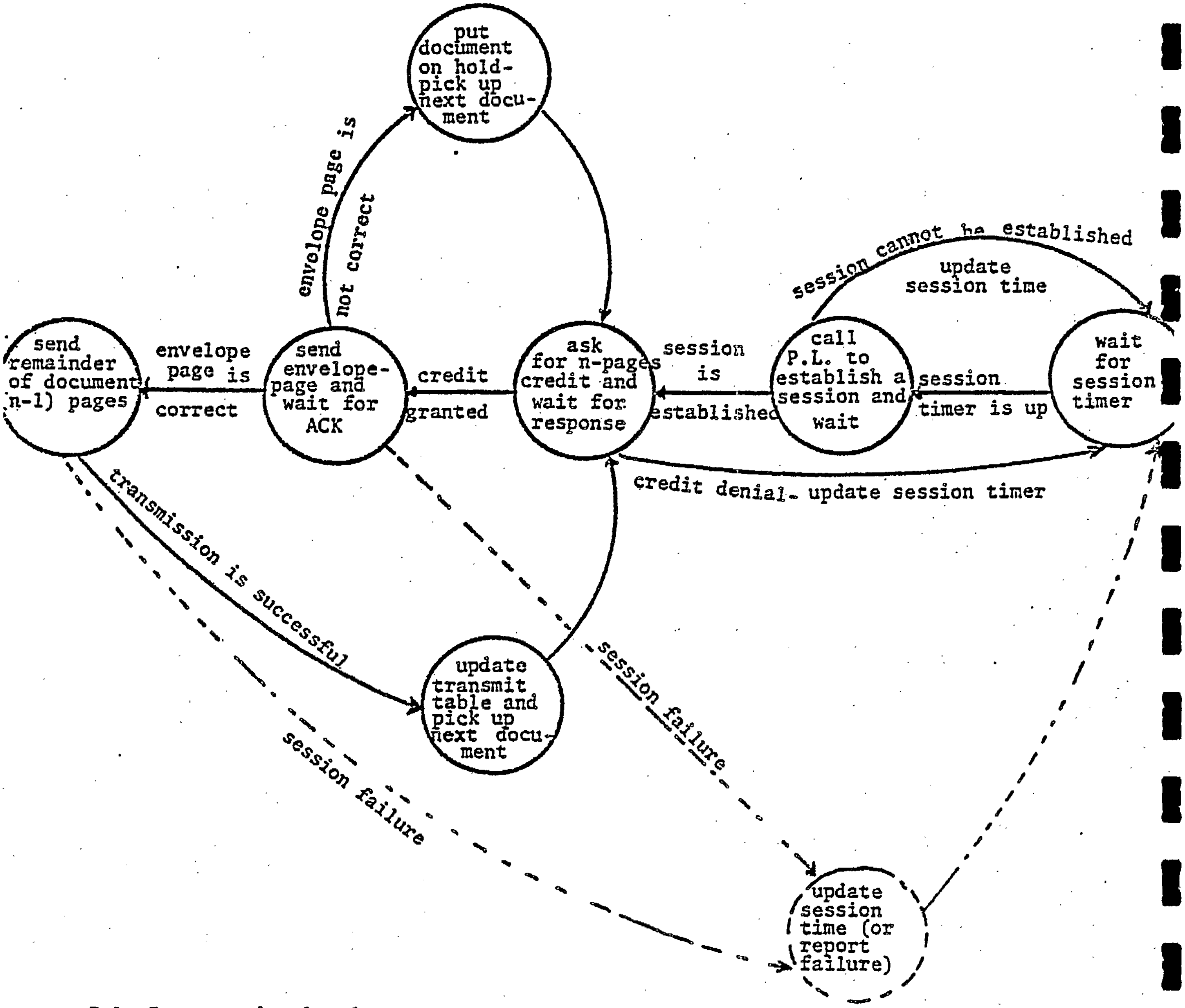
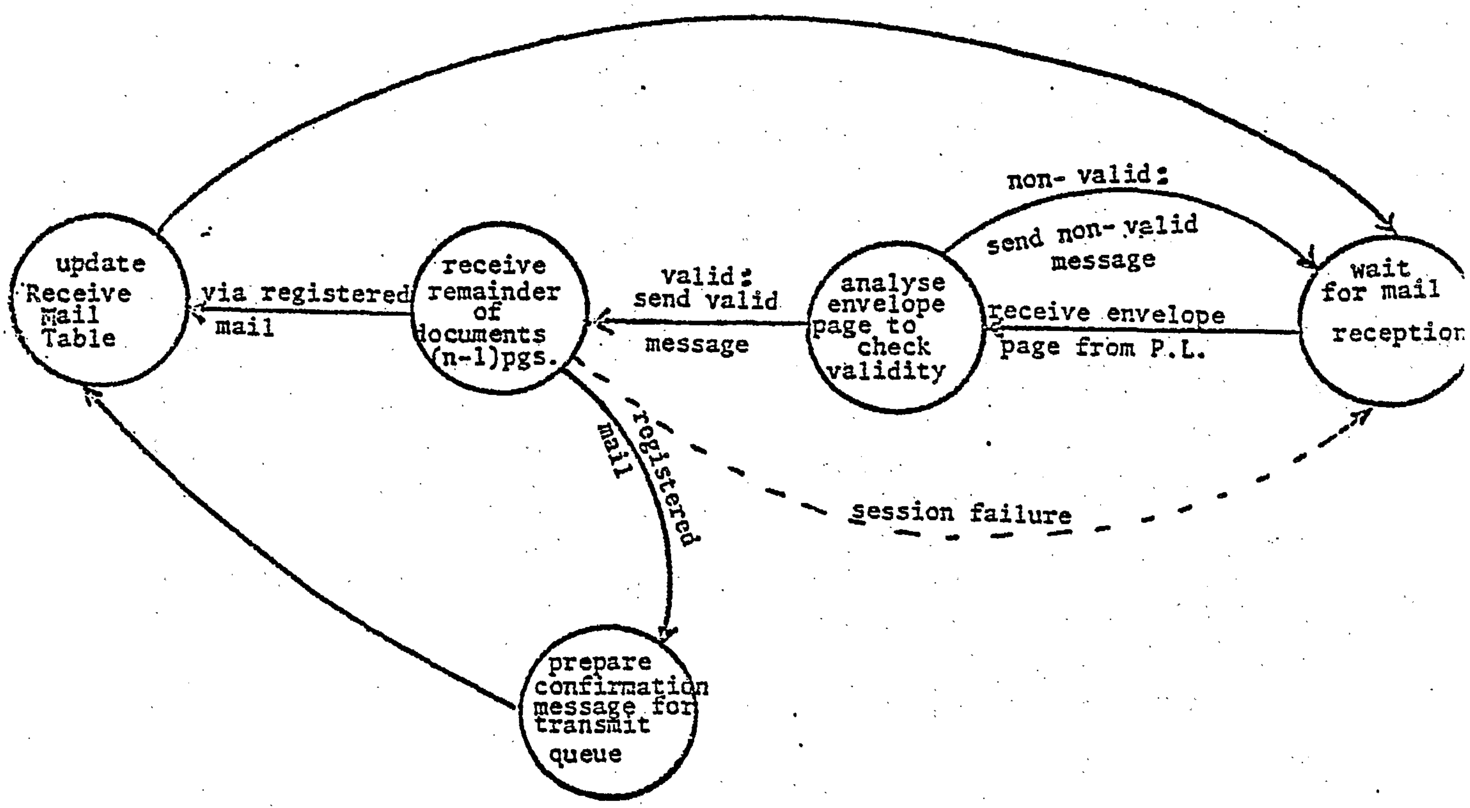


Figure 3.25: ELECTRONIC MAIL APPLICATION LEVEL MESSAGE FLOW



P.L.- Presentation Level Protocol

Figure 3.26: ELECTRONIC MAIL APPLICATION FUNCTIONS: (SEND MODE)



P.L. — Presentation Level Protocol

Figure 3.27: ELECTRONIC MAIL APPLICATION LEVEL FUNCTIONS: (RECEIVE MODE)

Chapter 4

Teletex

4.1 Introduction

The main purpose of this chapter is to examine, from the viewpoint of application-dependence, the proposed Teletex services and functions at the various levels of the open system model. The reader should note that the Teletex proposals have been evolving throughout the course of this study and that the comments on Teletex herein may therefore be outdated by the time this report appears.

Teletex is essentially a more intelligent version of Telex. For our purposes, Teletex is described in three draft CCITT recommendations. S.c [1], S.x [2] and S.d [3] which describe respectively the features of a Teletex terminal, the Network Independent Transport service for a Teletex terminal and the Session and Presentation services for a Teletex terminal.

Of particular interest is the situation where multipurpose terminals may connect to the Teletex service as both senders and receivers of Teletex messages. Because of the ubiquity of word processing terminals in business offices it seems logical to suppose that word processing terminals would be the first ones to offer this multipurpose capability. Figure 4.1 illustrates the desired capability.

The problems associated with Teletex, from this point of view, are related to its dedicated nature. A purpose of this chapter is to determine if generalizations are required to enable it to accommodate other applications such as interworking between generalized word processing terminals.

4.2 Teletex Levels

4.2.1 Introduction

The Teletex service provides for fixed format pages to be delivered to dedicated Teletex receivers. There is a limited range of document representation and a limited capability for negotiation on alternate character sets; embedded control characters are required for this purpose. However, in spite of the restricted aims of Teletex, the levels in the Teletex service correspond to those in the Open System Model and could easily be used for multiple purposes with proper definition. This section discusses the main issues associated with each level. Because the Teletex draft recommendation S.d suffers from the lack of clear definition of the interfaces described in Chapter 2, it is difficult to derive a definitive description of these interfaces using the notation presented in Section 2.2; accordingly, no such description is presented.

4.2.2 Issues at the Transport Layer and Below

Draft Recommendation S.x treats only the network, link and physical levels and leaves the transport level for further study. Accordingly, there is nothing to say here about this level.

4.2.3 Teletex Session Layer Issues

Teletex sessions may be conducted in various modes (one-way, two-way alternating, two-way simultaneous). Therefore, it is necessary to distinguish presentation commands from responses to allow responses to be sent by a terminal which is not enabled to send commands. Therefore presentation information commands and responses must be sent under cover of separate types of session control headers (CSI and RSI). However, the session level does not examine or act on any presentation information.

There is a presentation control parameter field specified in the session start command (CSS). However, its content and use are not specified. Possibly it could be used to negotiate character sets or translation tables to be used by the presentation level during the requested session.

The session start command (CSS) also provides a field for calling terminal ID. This field could presumably be used to provide the calling terminal's telephone number in the case where the lower levels do not provide it (consistent with this use in Chapter 3 for connecting to an X.21-like service). However it seems more appropriate to assume that the Transport level will provide the telephone number if the network level does not.

The session start command (CSS) has a mandatory "Service Identifier" Field which could possibly be used in a multi-purpose environment to request either a non-Teletex or a Teletex session. The specific use of this field is not described in the S.d document. However, its intent is probably to identify services within Teletex. If it is to be restricted in this way, then there is no provision for serial or parallel sessions for different applications. The session identifier field in the positive response message (RSSP) certainly does not help; it is used only to provide unique numbering of sessions.

If there can be no serial or parallel sessions for different applications, then a Teletex receiver must be dedicated solely to Teletex at the session level, because it must be continuously available for reception. A Teletex sender could conceivably switch between different session level modules for different applications (ready in memory or loaded on demand), using the special Teletex session level only when sending to a Teletex receiver. The consequences are a need for more than one session level package in any multipurpose terminal which includes Teletex transmission in its repertoire. Otherwise a multipurpose terminal cannot be used as a Teletex receiver.

A capability list command (CXCL), and an optional terminal capabilities field in the session start command (CSS) and in the session start positive response (RSSP) provide for hand-shaking on compatible capabilities for carrying on a session between two Teletex terminals.

The capability list command may be used for various purposes which are for further study according to S.d:

- specify receiver capabilities required;
- storage negotiation;
- use only at document boundaries or within document;
- flow control is an associated issue.

It has been proposed elsewhere that Teletex flow control be accomplished by:

- page-by-page acknowledgement ("send-and-wait") above the Transport Layer, or

- explicit use of Transport Layer services to disable and enable reception and transmission over a transport connection.

As indicated by S.d, this issue is apparently still not resolved.

The method of flow control by credit grant request and checkpoint status request used in Chapter 3 at the Presentation level in an electronic mail application appears to be suitable here also.

The imbedding of storage negotiation and flow control functions in the session layer, as suggested for the capability list command, appears to be an unnecessary complication. As described in Chapter 3, these functions were appropriately performed in the Presentation level or the lowest application sub-layer in MSDL implementations.

Checkpointing is, appropriately, left to the presentation layer, as it was in the MSDL applications described in Chapter 3.

4.2.4 Teletex Presentation Level Issues

The presentation functions defined in S.d are primarily concerned with delimiting documents and pages and with recovering documents and pages when errors occur. There is a flow control mechanism, complete with acknowledgement window, tentatively suggested, with the reservation stated that unless it is required for recovery, flow control is perhaps best left to lower levels.

There are, surprisingly, no translation or formatting services or functions specified at the Teletex Presentation level. And there is no encryption capability.

4.2.5 Teletex Application Level Issues

The Teletex Application level is not defined in S.d and is not explicitly defined in S.c.

We note in passing that there is apparently no provision for delivery confirmation of Teletex Application level messages to the intended recipient (as there is in the Electronic Mail system described in Chapter 3).

4.2.6 Text Translation/Formatting Services and Functions in Teletex

Text translation/formatting services and functions remain unspecified with respect to levels of the Open System Model. However the terminal functional specification S.c does provide a partial statement of the basic requirements.

There is only a limited capability in Teletex for negotiation on text translation. Draft recommendation S.c specifies use of escape sequences to move between the (default) G0 code set and the alternate G2 code set. There is also similar provision for escaping to a G1 set of which there may be many versions. Possibly the G1 version intended would have to be passed to the session establishment service of the Session level for inclusion in the presentation parameters field of the session start request message. However this is unclear. It may be intended that the Application level negotiate directly on code sets.

There is no mechanism defined for negotiating page formats.

4.3 Summary of Teletex Issues

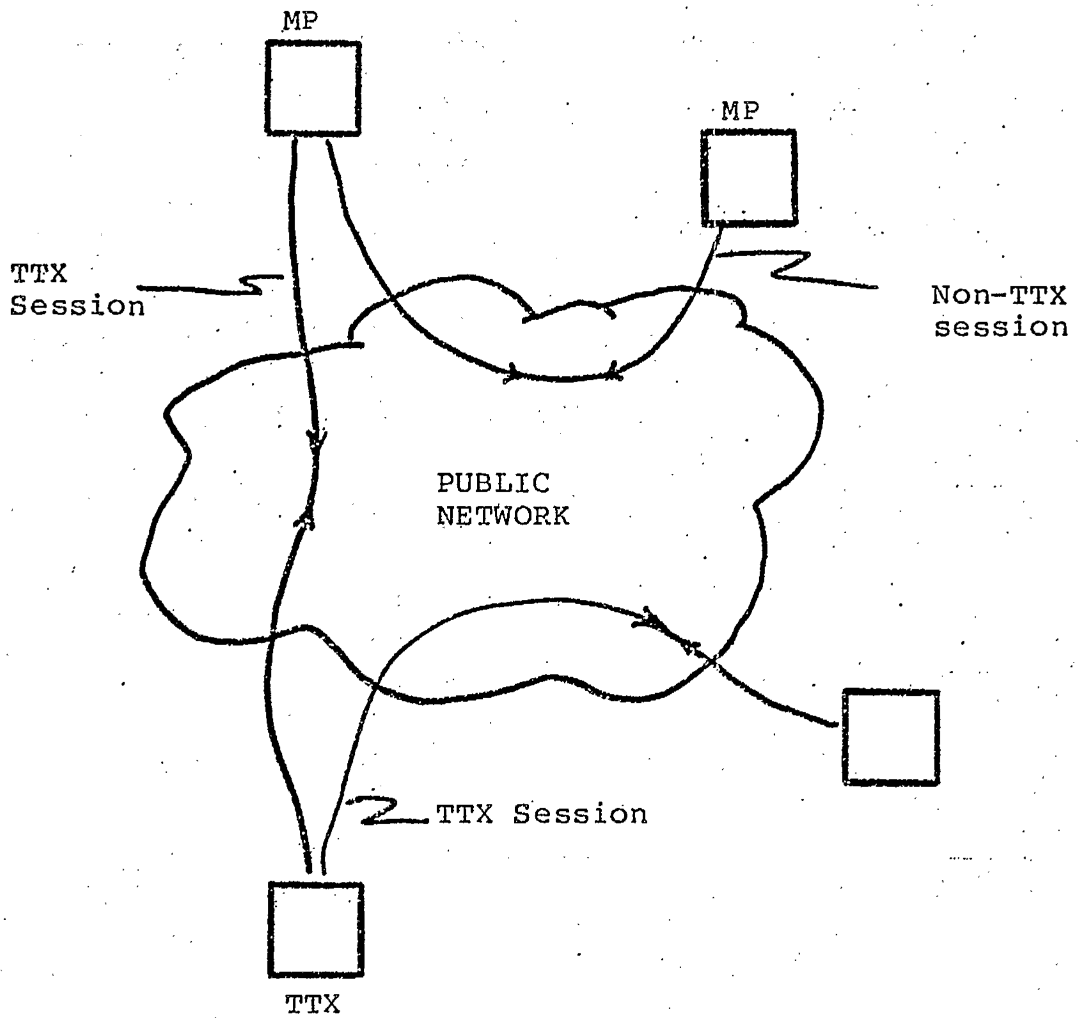
The main problems and issues identified in this chapter for Teletex have all been identified and discussed in Chapter 3 in relation to other applications. There is little new here for general discussion in this specific concext. Accordingly, further general discussion is postponed until Chapter 7.

4.4 References for Chapter 4

1. "Draft Recommendation S.c (TELETEX Terminal)," COM VIII - No. 81-E, November 1978, CCITT.

2. "Proposed Text for Proposed Draft Recommendation S.X," January 1980.

3. "Proposed Text for Draft Recommendation S.d," January 1980.



LEGEND

TTX - Teletex
MP - Multipurpose

Figure 4.1: Teletex and Multiple Purpose Terminals

5.1 Introduction to Facsimile

5.1.1 Existing and Future Facsimile Machines

Growth of installed facsimile systems has mushroomed in the past ten years and it is estimated that more than 200,000 facsimile units are in use at the present time. The widespread acceptance of facsimile services has been held back by the lack of compatibility between machines from different manufacturers, high cost, low speed, and mediocre copy quality. However, these drawbacks are slowly being overcome. The advances made in the microprocessor field and in communications' technology promise economically feasible solutions to the incompatibility problems among dissimilar facsimile machines. In addition, equipment rental costs are dropping and several machines now provide for automatic operation to conserve personnel costs and to permit transmission during off-peak hours when line charges are low.

Several facsimile machines have been recently introduced with 30 seconds, one or two minutes transmission speeds. Such machines are bound to reduce further transmission costs. In addition, copy quality is being improved by the development of better printing devices, improved modulation techniques and upgrading of telephone line qualities.

Facsimile machines are divided into two broad classifications - convenience and operational. Convenience machines are generally low in cost and slow. Typically, these machines are used for sending between 25 to 50 copies per month. Because of this light use, buyers of convenience machines are sensitive to monthly leasing costs, but not overly concerned with additional costs per copy (paper plus transmission charges.)

Operational machines are used for high volume transmission and individual copy cost are of prime importance. Most of these machines provide high speed transmissions in addition to other advanced features such as automatic document-loading, automatic dialing and answering, interrupt capability and multiple-document handling.

Table 5.1 provides a summary of the major characteristics of a sample of Facsimile machines marketed by different manufacturers. The table indicates the variations in the speed, resolution, price and other features of different facsimile machines.

	QWIP	QWIP	XEROX	XEROX	PANAFAX	PANAFAX	GRAPHIC SCIENCES	3M	RAPICOM
MODEL NUMBER	1200	2000	X410	TC-200	MV-1200	UF-20	DEX-700	Express 9600	100
RESOLUTION	96x96	100x100	96x96	96x96	200x200	200x200	176x176	200x200	200x200
SPEED (MIN)	6.0	2.3	6.0	3	3.6	0.4-0.8	2-3	1.7	1.5
MULTIPLE DOCUMENT FEATURES	NO	NO	NO	YES	OPTION	OPTION	NO	YES	OPTION
AUTO-DIALING	NO	NO	NO	YES	NO	NO	NO	NO	NO
AUTOMATIC ANSWER	NO	NO	NO	YES	NO	NO	NO	NO	NO
INTERRUPT	NO	NO	NO	YES	YES	YES	YES	YES	YES
PRICE ($)	1200	1900	1200	8900	4500	12500	4750	14500	

Table 5.1: <u>Comparison of Facsimile Machines</u>

5.1.2 Standard and Compatibility Issues

The areas of incompatibility among facsimile machines fall into the following functions:

i) Document Scanning Techniques

Several scanning techniques are currently employed: Rotating Cylinder, Rotating Helical Aperture, Rotary Scan Head, Multiple Photo Diode Array, CRT Flying Spot Scanner, and Laser.

The result of the scanning step can be either an analog signal or a digital signal. Thus, scanning methods are generally classified as analog or digital. Analog scanning techniques cover several points in the colour spectrum between black and white. Thus, they are capable of preserving different gray shades that may exist in the document.

Digital techniques represent all shades in the document as black or white. The information on the document is represented in these techniques as a continuous stream of binary digits.

The quality of the copy being produced and transmittd depends on the resolution used as well as the scanning method. High quality machines tend to generate more (analog or digital) transmitted traffic per document compared to lower quality machines.

ii) Modulation Techniques

The signals resulting from the scanning process are modulated before transmission over the communications channel (telephone lines, satellite channels, micro-wave circuits, etc.). Several modulation techniques are currently employed (AM,FM,PM, etc.). For two facsimile machines to communicate directly over a channel, the modulation technique used in each has to be matched to the other.

iii) Compression Schemes

Both analog and digital types of scanning and transmission techniques are amendable to data compression whose aim is to reduce the large volume of traffic generated by scanning a document. Generally, digital machines can provide greater compression than analog ones. In addition, the recovery of a modulated digital signal is less sensitive to noise and signal fades (and thus is less prone to error conditions) than a modulated analog signal.

iv) Handshaking Routines

These include the method of synchronization between two machines, the codes (or signals) used for formatting, error detection procedures, start and termination of document transmission.

v) Scanning and Transmission Time

The total time required for the transmission of a document is a function of the scanning speed, the resolution used, the modulation technique, the compression ratio and the speed of the communications channel. The first four factors vary in different facsimile machines and as a result, different transmission times exist. Machines with different speeds cannot, however, communicate to reproduce a document without substantial buffer storage.

The above incompatibility factors have until recently mitigated against network-wide communications among users of different facsimile machines. However, as it has been with the computer industry, smaller manufacturers are now building their machines to be compatible with those being produced by the industry leaders. While this trend is expected to alleviate the incompatibility problems to some extent, a complete solution can be arrived at only through the development of widely acceptable standards and the utilization of recent advances in the technology of computer-communication networks.

As a step towards developing standards in the facsimile industry, the facsimile committee in CCITT has classified four groups of facsimile machines:

Group 1 - This is similar to the Xerox Telecopier II and the Magnavox-850 machines. It transmits 4 and 6 minute facsimile data using a well defined protocol and FSK (FM) low-speed transmission over telephone lines.

Group 2 - This standard is mostly with European PTT and adds a 3-minute capability. Transmission is AM.

Group 3 - The first modern standard. These machines would use some run-length encoding scheme for data compression. High-speed transmission of data is expected to reduce substantially total document reproduction time.

Group 4 - Super-high-speed equipment. Standards are to be defined.

As stated earlier, while such standardization efforts represent useful steps towards interfacing machines produced by different manufacturers, they are not, however, sufficient to allow for general networking capability involving various facsimile systems. The main reasons are:

i) Only machines within a group can communicate directly. A machine must be equipped with all groups in order to be totally compatible. The price of such a "super" machine would be prohibitive.

ii) The compatibility defined for all groups is lost on machines which use a more efficient proprietary communications technique such as Graphic Sciences' "DEXNET", Panafax's FAST SPEED, Xerox's TC200 and the entire RapiCom network. Such systems are widely used at the present time and a general networking service cannot afford to exclude their users since they form the bulk of the network's potential customers.

The preceeding discussion illustrates the need for developing networking approaches to augment the standardization efforts towards finding a complete and economically feasible solution to general facsimile intercommunications.

5.2 Existing and Future Communication Systems for Facsimile

In order to utilize facsimile to its full potential, a concept of networking must be developed. A number of attempts at networking have been made with various degrees of sucess and limitations. These attempts have resulted in services being offered at the present time. In the following, we discuss the nature of these services, their advantages and limitations. The discussion will serve as a preamble to the application of the Open System Interconnection Concept to future facsimile networks.

5.2.1 Existing Facsimile Networks

Existing facsimile communication systems can be classified according to the compatibilities of the machines interconnected in the network.

i) Communications Among Compatible Machines:

Communications among machines from the same manufacturer is usually carried out over public telephone (voice grade) networks. The same is expected to apply to different machines within the same group as classified by the CCITT facsimile committee.

The main advantages to this approach are that the network can be developed in a piecemeal manner without very much planning and can be expanded as the traffic builds. The major disadvantage is that the user is limited to the created network and cannot communicate with other machines outside its standardized group. Also, as pointed out earlier, this approach excludes machines which use a more efficient proprietary communications techniques.

ii) <u>Common Carrier Service Networks</u>

Common carriers began to offer facsimile sercices to data communications users in 1975. The first comprehensive facsimile service was offered by Graphnet Systems Inc. Information from facsimile machines can be transmittd to other compatible or incompatible machines. Messages sent on Graphnet are digitized, compressed and converted to the appropriate output format and speed by special purpose processors. The service uses the switched telephone network and thus reaches almost all locations on the continent.

Recently, a new facsimile service named Faxpac has been announced by ITT. Faxpac is a store-and-forward network customized to handle facsimile message communications. The network provides single terminal-to-terminal delivery as well as message broadcasting to multiple terminals. The network also provides for fax-to-fax communications among incompatible machines through digital conversions of facsimile signals and data at the network's main switching nodes.

Similar services are expected to be provided soon by other carriers such as Satellite Business Systems, Western Union, American Satellite, Data Transmission Corp. and AT&T's Advanced Communications Service.

The proliferation of such "standard" facsimile service networks will help in resolving the incompatibility problem to a large extent. However, two serious problems persist:

1) Subscribers of a given network cannot communicate with those of other networks. To solve this problem, an approach for internetworking must be developed. Past difficulties that have been encountered in interconnecting computer-communications networks are bound to hamper a similar attempt in the facsimile world.

2) The international tariff structure presents substantial cost problems when using any of the facsimile service networks. In addition, users with a particular facsimile processing needs will find it difficult to configure their system in a way that minimizes their service cost. For example, a user with a cluster of several slow speed facsimile terminals may be at a disadvantage when using a network with a tariff structure based solely on machine line connect time.

The Open System Interconnection Concept presents an alternative approach which promises to overcome the above explained difficulties. This concept is explored in the following.

5.2.2 Feasible Facsimile Configurations in the Open System Interconnection

It can be concluded based on the previous discussion that the "ideal" facsimile network should be able to select and optimize the following parameters:

- Variations in Facsimile Traffic Volumes: The traffic generated from user centres range from infrequent and low volumes to continous and high volume. The user should have the flexibility of configuring his service centre to match his loading conditions and service speed requirements.

- Support of Incompatible Machine Types: This means removal of all the constraints that limit the interface between incompatible facsimile machines.

- Variations in Tariffs: Tariff structures differ at the international level and according to the specific communications service provided. It should be possible to take these variations into consideration and give the users the required flexibility to attain optional transmission routing.

- Multi-Copy Capability: In many applications there is a need for a broadcasting capability to multi-receiver terminals. The network should be able to provide this service without image degradation.

- Modularity: It should be possible to expand the network both at the user's node level and at the network support level without major equipment replacement or intensive labor involvement.

- Flexibility: It should be possible for the network to accommodate several data compression schemes and communication codes. This will help in optimizing the performance of the network for a mix of facsimile application. It is also desirable to have a built-in flexibility to accommodate future technological advances such as optical-character recognition facsimile systems that can transmit information to terminals for print-out or video reproduction.

The Open System Interconnection approach can be considered superior to other approaches if it provides the facility to select and optimize the above parameters. To examine this possibility, we proceed in two steps. First, we define three general configurations that can be combined in a global network to interconnect facsimile machines of all possible types and CCITT groups. Second, we examine in some detail the structure and functions of the Open System Interconnection Protocol levels as applied to facsimile. The first step is presented in this section while the second step is covered in the remaining sections of this chapter.

We define three configurations for interfacing facsimile machines to a network with an Open Interconnection Protocol. The three configurations can co-exist in any combination in the network.

Configuration #1: Facsimile Service Node: In this configuration, a relatively large processor is used to interface a number of different facsimile machines to the network (see Figure 5.1). The Open System Interconnection Protocol is executed by the processor. Each facsimile machine communicates with the processor using dial-up telephone service or any alternative local service. The main features of this configuration are:

- Facsimile machines expected to be connected in this configuration are Group I machines, low volume Group II machines and isolated (non-clustered) Group III machines. The Configuration can also support machines outside CCITT group classifications.

- The processor has a mass (disk) storage to pool all facsimile data converted into digital format during transmission.

- The maximum number of facsimile machines served by a node of this configuration is determined as a function of the capacity of the port connected to the processor and the desired blocked call probability.

- The machines can function in the attended or unattended mode.

Configuration #2: Local Concentrator Configuration: In this configuration, a cluster of facsimile machines is grouped and served by one processor (mini-computer size), as shown in Fig. 5.2. The processor executes the Open System Interconnection protocol and thus acts as a concentrator to convert the cluster of machines into the network. The machines are wired directly to the processor's interface. The communications between the machine and the processor may be based on a polling protocol in which the processor acts as the polling controller. Alternative local communication protocols can be employed if deemed suitable to match the traffic pattern.

The main characteristics of this configuration are:

- Facsimile machines expected to be connected in this configuration are high volume Group II machines and group III machines.

- The processor has a mass (disk) storage to pool all facsimile data converted into digital format during transmission..

- The machines function in the unattended mode.

- The power and storage capacity of the processor are selected to handle the traffic generated from the local cluster of machines as well as the traffic received from remote machines.

Configuration #3: Facsimile Front-End Configuration: A front-end processor is used to connect a single facsimile machine into the network. The facsimile machine is of the digital type with high speed and high volume activity. This type of machine represents Group IV machines that are currently being examined by CCITT for standardization. The processor has a limited disk storage (500 K bytes- 1M bytes) which buffers the outgoing data before transmitting it. The storage is needed to allow the machine to communicate with other slower speed machines.

This configuration can also be used to upgrade the capability of a Group I machine or a Group II machine to a modern standard. The front-end-processor will perform all the necessary hand-shaking with the host machine, convert its facsimile signals to a digital format, execute efficient compression procedure and then transmit the information over the network. As well, the front-end-processor receives facsimile data destined for its host machine, performs the necessary conversions and deliver the signals to the machine for reproduction of the printed document.

Upgrading slow speed analog machine to a modern standard has been demonstrated recently to be commercially feasible. Compression Labs Incorporated (CLI) introduced a product, Fax-Comp, which can work with the low speed analog facsimile machines and provide data compression that is up to 5 times more effective on alpha/numeric text than the conventional run-length-coding scheme of the digital facsimile machines. The total cost of a typical low cost analog machine and a Fax-Comp remain competitive with the cost of a high speed machine with a comparable effective transmission speed.

Details for Interconnecting the above described configurations in an Open System Interconnection Protocol are presented in the following sections.

5.3 Transport and Lower Levels

5.3.1 Transport Layer in Facsimile

The transport layer main function is to provide transparent transfer of data between session entities.

In facsimile, the transport layer is expected to provide the following two types of services to the next higher layer:

(i) A connection oriented service: This will allow the transfer of one or more facsimile documents between a sender and a received facsimile machine.

(ii) A broadcast oriented service: This will allow the transfer of facsimile data from one sender machine to several recipient machines.

The general form of the first type of service has been analyzed by ISO (see reference [1]). The second type of service has not been defined yet by ISO and is currently under study.

An important function of the transport layer is to optimize the use of the available communications resources to provide the performance required by each transport user (session level process) at minimum cost. This optimization capability is particularly vital to the facsimile application where it is necessary to have two modes of operation:

(1) Direct Fax-to-Fax

For low cost, low speed, identical facsimile machines with low utilization, the minimum cost communications configuration may be attained by allowing the two machines to communicate directly over (dial-up) voice grade telephone networks. In this case, the open system interconnection role will be confined to perform the initial session set-up between the sender and recipient sites and then allow the two machines to communicate directly with each other (communications here is based completely on analog signals). At the end of transmission, the Open System Protocol will close the session.

(2) Common Format Communications

For the general case of incompatible machines with moderate or heavy utilization, the most cost-effective communications may be attained by using public data networks (packet or circuit switching networks). Here it is assumed that the facsimile signals are digitized and thus subject to compression and formatting under the control of the digital processors hosting the facsimile machines.

The transport layer in facsimile should thus be capable of handling each of the above two modes of operation. The second mode, however, may require more than one class of service as depicted by Figure 5.4.

The phases of operation within the transport layer are:

- establishment phase
- data transfer phase
- termination phase

We examine now the execution of these phases for the two modes of operation.

(1) Direct Fax-to-Fax Communications

The decision to use this mode of communications is done at a higher protocol level (presumably at the application level or at the presentation level). The transport level gets the request from the session level to establish a transport-connection with another transport-address associated with a correspondent session-entity.

The transport layer needs to determine the network-address identifying the transport-entity which serves that correspondent transport address. Because transport-entities support services on an end-to-end basis by means of end-to-end functions, the network-addresses on which the Transport Layer maps transport-addresses are those identifying the end transport-entities. In a dial-up service, the network address translates into a telephone number. The protocol layers below the transport level can be based on an HDLC type of protocol.

Following the establishment of the transport connection, the high level protocols communicate to identify the sender and recipient facsimile machines. Each end will perform initial handshaking between the facsimile machines and the host processor, the transport layer is then instructed to turn over the connection to the facsimile machines so that they can communicate directly. Switching the connection to the machines takes place at the physical link level. Facsimile data will be transmitted as analog signals directly from the sender machines to the recipient machine.

When document(s) transmission between the two facsimile machines is completed, the fax machine's tones signalling the end are picked by the physical link ends and the higher level protocols are notified to take steps to close the session.

Figure 5.5 illustrates the role played by the physical link protocol. Initially the physical link is used by higher level protocols to set up the channel between the two facsimile machines. The physical link protocol is invoked once again when the transmission is completed or in the event of a facsimile machine malfunctioning.

(2) Common Format Communications

In this mode of operation, facsimile data is converted into digital formats which are stored and manipulated by the host processor. Pages of a document are thus transmitted from sender to recipient as described in the Electronic Mail Application. The transport and lower level layers are thus similar to those used in the Electronic Mail Application, described in Section 3.3 of this report.

5.3.2 Summary of Issues at Transport and Lower Levels

Several issues remain to be addressed with respect to the Transport and Lower Level Protocols in Facsimile:

- Broadcast-oriented services at the Transport Level have not yet been defined by ISO. Yet this service is vital in Facsimile to allow a single sender to broadcast the same document to multiple sites.

- In the direct Fax-to-Fax communications mode, the transport level is expected to set up connections with three phases: Establishment Phase, Data Transfer Phase and Termination Phase. The first is initiated based on a request from the session level. The second phase involves switching the communications channel (at the physical link level) to allow the two Facsimile machines to communicate directly. The practical feasibility of this phase is still open to question. Finally the execution of the termination phase requires signalling from the physical link level; an aspect which requires further scrutiny. Failure to detect the termination tones may leave the connection open for an indefinite period of time.

- The choice of which service mode to use (common format or direct Fax-to-Fax communications) has to be decided at higher levels since it is dependent on the types of Fax machines used at each end. Thus the higher level protocols are not network independent in the fullest sense. The alternative would be to leave this decision within the Transport Level, which would tend to complicate its structure.

- Error discovery depends to a large extent on the adequacy of the interface to the facsimile machines and the completeness of the status signal set of these machines. In the direct fac-to-fax mode, the Transport Level does not control data transmission and will thus be unable to check service quality and possible errors. It is essential to examine the set of status signals of each facsimile machine to ensure that the Transport Level is constantly aware of the status of transmission and can inform the higher level protocols of any disorder.

5.4 Session Level

5.4.1 Sessions in Facsimile

Two types of sessions are needed for the facsimile application:

(1) Session in direct Fax-to-Fax Machine Communications:

In this mode of operation, the request to establish a session will be received from the presentation level protocol. The session establishment protocol will be identical to the protocol discussed in Section 3.4.

The following actions will follow the session establishment:

- the presentation level is informed of session establishment
- the presentation level requests the session level to start transmitting facsimile data.
- the session level requests the transport level to connect the sending and receiving machines directly.

Once the two machines are connected at each end of the line, facsimile data in the form of analog signals will be transmitted directly between the two machines. All the protocol levels above the physical link level, including the session level protocol, will play the role of "observers" until transmission is completed or a channel failure is detected.

The session is closed once the transmission is completed. The protocol described in Section 3.4 can also be used for terminating a session. Figure 5.6 illustrates the functions of the session level protocol in this mode of operation.

The following remarks apply to this mode of transmission:

- In the open system interconnection, a presentation level entity is capable of aborting a session (a facility provided for flow control and to cope with serious failure conditions). In this mode of operation, the presentation level protocol can request the session level to abort a current session if it receives a message indicating failure of the facsimile machine it is hosting.

- Units of data here (in the form of analog signals) do not actually pass through any of the protocol levels above the physical link level. Thus the session level protocol acts only as an observer while data transmission is taking place.

- No roll back and recovery is possible in this type of session since no signal storage is taking place. An interrupted session will have to be reset and all previously transmitted data will be discarded at the receiving end.

(2) Sessions in the Common Format mode of Communications:

This mode of transmission resembles to a large extent the transmission in electronic mail systems. Thus session establishment, data exchange and session terminator's protocols can be identical to those protocols described in Section 3.4 of this report.

5.4.2 Summary of Issues at Session Level

Several issues require further investigation in the session level protocol of facsimile:

- It appears that two different types of sessions are required for the facsimile application: (1) Sessions involving direct facsimile machines communications, and (2) Sessions in which facsimile data is transmitted in the form of data messages between higher level protocols. The decision regarding which session type to select depends on the type and compatibility of the communicating machines. Since this information is available at the application level, it seems that the decision will have to be made at this higher level. This naturally violates the concept of independence of the session level from the levels above it.

- In the direct fax-to-fax mode of operation, several functions executed at the session level, such as session reset and session close, are triggered by signals received from the facsimile machines. Then signals are received at the physical link level and must be propagated to the layer above until they reach the session level. This means that certain mechanisms must be provided to ensure that the status of the channel is monitored continuously. The absence of such mechanisms can lead to some sessions being opened indefinitely while no data is being transmitted between the end nodes.

- Sessions for the broadcasting mode need to be defined. It is anticipted that a number of problems will arise when attempting to establish and maintain concurrent sessions with several recipient stations. Examples of such problems include the handling of one or more sessions being interrupted due to failure, roll back and recovery for such sessions and the role played by the session level in managing quarantine units of data.

5.5 Presentation Level

5.5.1 Presentation Level Functions and Messages

In the open system interconnection, the presentation level protocol is concerned with the management of formats and the management and performance of transformations for the users of the architecture.

In facsimile, the presentation level protocol performs the following functions:

- initiate and terminate sessions
- negotiate facsimile compression and formatting procedures
- transfer of data and flow control

We have already idenified two modes of communication in the Open System Interconnection in the facsimile application:

(1) Direct Fax-to-Fax Communications

In this mode, the fax machine will communicate first with the local processor (the service node processor as in Figure 5.1, the local concentrator as in Figure 5.2, or the front-end processor as in Figure 5.3). The local processor will communicate with the corresponding remote processor by establishing a session, presumably over a slow speed (dial-up) line. Since the two facsimile machines are compatible, there will be no need for document formatting or compression. Each processor will perform the necessary hand-shaking procedure with its local fax machine. The sending processor will then send a "ready to transmit" signal to the recipient once it completes the hand-shaking procedure. The recipient will respond by sending a "ready to receive" signal to the sender, which indicates that it has also completed the hand-shaking procedure. Following the exchange of these messages, the two fax machines will then be connected at each end of the line and can communicate directly until the transmission is completed. When the two machines signal end of transmission, the signal is picked up by the local processor at each end. The session will be terminated at this point.

The hand-shaking procedures between facsimile machines can either be

a) according to CCITT specifications T-30

or

b) machine specific hand-shaking procedures; the machines are identical in this case

Figure 5.7 illustrates the basic role of the presentation level protocol in this mode of transmission.

(2) Common Format Communications

This mode of communications is employed for the general case when one or more of the following conditions exist:

(i) the two facsimile machines are not compatible
(ii) transmission takes place over public data networks with multiple recipient sites requirement
(iii) there is a need for data compression to minimize and/or to speed up data transmission

Each facsimile machine will transmit its fax data to the host processor which connects it to the network. If the machine is sending analog signals, the host processor will transform it into digital formats. The fax data will then be compressed and converted to the common network format to be transmitted over the network. The fax data in the common network format will be received at the remote end and converted to a format compatible with the local recipient machine.

Each host processor is equipped with a secondary storage device whose capacity depends on the volume of data handled by the processor. Transmitted and received fax data are stored on the device for various durations of time for the following reasons:

(i) to cope with the speed difference which may exist between the sender and the recipient machines

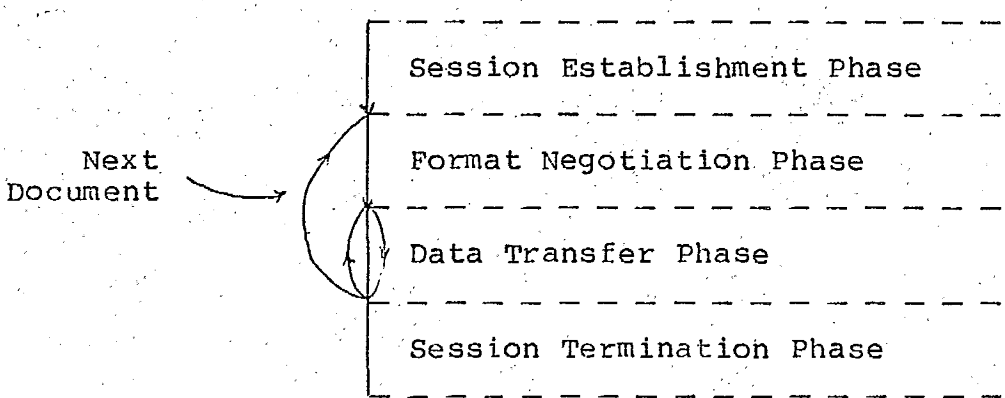
(ii) to provide partial back-up in case of channel failure which can be restored without the need to retransmit the entire document, i.e., in cases where the roll-back involves few lost pages

(iii) to allow reformatting and compression/expansion of facsimile data

The presentation level protocol can be viewed to consist of four phases:

(i) session establishment phase
(ii) format negotiation phase
(iii) fax data transfer phase
(iv) session termination phase

The sequence of execution of these phases is depicted in the following diagram:



(i) Session Establishment Phase

The presentation level receives a message from the delivery level to start a session. The presentation level requests session establishment from the session level protocol by passing to it the address of the recipient station. The session level responds either by confirming "a message has been established" or a "session denial". If the former is received, the presentation level proceeds to the negotiation phase.

(ii) Format Negotiation Phase

The sender processor transmits a credit request message to the recipient host. The message contains:

- the sending machine type
- the recipient machine I.D.
- the expected number of pages in the document (if known)

The recipient processor decides whether to grant the credit or not based on two factors:

1) availability of storage space locally. The required space is calculated as a function of the difference of speed between the two machines

2) the local (recipient) fax machine is ready to receive, i.e., it is responding to a polling or dial-up signal from its host processor

Following the receipt of a credit grant message, the sender processor sends Format proposal message. This message contains:

- the compression procedure
- the format and control sequence to be applied

The recipient host responds by a Format acceptance message. The sender host executes hand-shaking procedures with the sender machine and sends a ready to transmit message to the recipient processor, which responds by performing hand-shaking procedure with the recipient machine and sends a ready to receive message to the sender host.

Following the ready to transmit and ready to receive messages, document transmission begins.

(iii) Data Transfer Phase

The basic data unit transmitted under the presentation level protocol is document page. Pages are numbered and transmitted sequentially in the negotiated format. At end of page and end of document control bit sequences are used and are implied in the format definition.

Transmission of document pages may be interrupted as a result of any one of the following two events:

- line failure: temporary disruption of communications may result in loss of some facsimile data. If communication resumes before the session is terminated, the sender host transmits a check point message to the recipient to get the number of the last page received. The sender then retransmits the missing pages and continues until the entire document is transmitted.

- Recipient in trouble: If the recipient host detects fast accumulation of data and is running out of space, or that its recipient fax machine is malfunctioning, it requests the sender host to halt transmission by sending a "stop transmission" message. In case of a total breakdown of the recipient fax machine, the recipient host will transmit a terminate transmission message to the sender host which promptly terminates the session.

(iv) Session Termination Phase

The presentation level instructs the session level to terminate an active session if any of the following events occur:

- last document in session has been transmitted. The presentation level receives a signal from the sender machine to indicate that no more documents are ready for transmission.

- a major failure in the communications channel as indicated by a mismatch in check point messages.

- credit denial message is received in response to credit request.

- no response is received for a "ready to transmit message".

- failure (breakdown) of local or remote facsimile machines (if the remote fax machine breaks down, a terminate transmission message will be sent from the recipient to the sender).

The Presentation Level Protocol messages are summarized in Table 5.2. The last three messages in the table for the rcipient side are introduced for the following reasons:

<u>Stop Transmission</u>: is sent from the recipient to the sender to halt transmission temporarily

<u>Resume Transmission</u>: is sent following a Stop Transmission Message

<u>Terminate Transmission</u>: is sent from the recipient to the sender to request session abortion (the document whose transmimssion was aborted must be transmitted in full at some later time

Figure 5.8 summarizes the basic functions and interfaces of the presentation level protocol. Figure 5.9 illustrates the execution sequence of these functions.

Table 5.2: Summary of Presentation Level Messages

Sender	Recipient
- credit request	- credit grant - credit denial
- format proposal	- format acceptance - format rejection
- ready to transmit	- ready to receive
- check point	- check point response
	- stop transmission - resume transmission - terminate transmission

5.5.2 Issues at Presentation Level

Three important issues remain to be addressed in the Presentation Level for facsimile:

1. The large volume of data generated in facsimile requires vast secondary storage, especially in situations involving a large speed difference between sender and recipient fax machines. In this case, it is essential for the host processor to be able to interrupt the sending machine for any duration of time, followed by resumption of operations. The interrupt capability may be also needed in case of temporary failure of the communications link. The interrupt capability, however, is not a feature available on every fax machine (see Table 5.1).

2. The broadcasting of fax data to multiple recipient sites can be accomplished by setting up concurrent sessions between the sender and the recipient sites. The presentation level protocol can then be executed simultaneously with each site. However, if any of the recipient machines is much slower than the sender, it will be the responsibility of the recipient machine to provide enough space to buffer the data which is arriving at high speed but being delivered to the recipient at slower speed. The recipient host takes this difference into account at the time of exchanging the "credit request" and "credit grant" messages.

3. One issue that must be resolved in the broadcasting to multiple sites is the action that must be taken by the sender host after it receives a "stop transmission" message from a recipient site which is facing a storage overflow problem.

5.6 Application Level

5.6.1 Description of the Application Level

The Application Level in facsimile will perform the following functions:

- Interface the sending user with the facsimile system to enter the data necessary for addressing and authorization. Also the interface between the facsimile machine and the host processor is managed by the delivery level.

- Reading and delivery of the facsimile document to the recipient(s).

Interface between the Sending User and the Facsimile System

To send a facsimile document, the user must enter three data values:

1. Recipient Station Number: This number identifies the network node which serves the recipient machine. The node can have any of the three configurations explained earlier.

2. Recipient sub-station Number: This number identifies the recipient's machine number in case more than one machine is being served by the node, as in configurations I and II.

3. Sender Authorization Number: This number is used for accounting and bookkeeping purposes.

It is possible that the first two numbers be lumped together to form one number which identifies the station number (as the significant digits) and the substation number (as the least significant digits).

These data values are entered using the telephone dial (as in Configuration I) or using a special touch-tone pad (as in Configuration II and III). For multiple recipient transmission, several recipient station numbers have to be entered in a sequence.

Before entering these data values, the user must ensure that his facsimile machine is on and the document is ready for scanning. The sending user role ends at this point.

Routing and Delivery of the Facsimile Document to the Recipient

Sorting and delivery of facsimile documents to the users at the recipient end are for the most part manual processes. This is because the scanned data in documents are not machine analyzable and the technology to do so is not being considered for the foreseeable future.

The CCITT Facsimile Rapporteur's Group defines the composition of a facsimile document as follows:

> "Every facsimile document shall comprise a preamble, and address a message"

The preamble consists of:

- name of accepting office
- identifying information (acceptance and/or sending number)
- indication of the number of messages to be transmitted
- date and time handed in for transmission
- name of sending office
- service indications of any

The address consists of:

- the designation of the addresses
- the detailed address
- other information needed for delivery of documents to users who are not co-located with the facsimile machine centre.

The message is the body of the document itself, i.e., all printed matter, drawings, etc., which form the subject of the facsimile document.

For sorting and delivery purposes, the preamble and address information will be included in the first page of each transmitted document.

5.6.2 Issues at Application Level

The following Application Level issues remain to be investigated in facsimile:

1. The addressing information in facsimile which is used to identify the recipient to the Open Interconnection Protocol has to be submitted separately, i.e., not through the fax machine, to the system. This limitation is due to the fact that facsimile data is not analysable by the digital processor. However, it gives the user interface a special significance in the sense that two separate actions have to be performed: getting the fax machine ready to start document transmission, and entering the addressing information through a different interface to the digital processor. Special attention must be given to the requirement of document broadcasting to multiple locations, particularly in Configurations I and II, to define how multiple recipient addresses can be entered.

2. Addressing data values (station and substation numbers) must be standardized to a universally acceptable format. As well, there is a need for defining the format of the first page of the document which contains the information required for sorting and delivery of facsimile documents.

3. Several delivery aspects remain to be defined: procedures for handling rejected documents, handling requests for retransmission of documents received with unacceptably low quality, and handling registered documents, i.e., documents for which acknowledgement of receipt is expected.

5.7 References for Chapter 5

[1] ISO/TC97/SC 16 N-117, "Open Systems Interconnection," November, 1978

[2] R.F. Sproull and D. Cohen, "High Level Protocols," Proc IEEE, Vol. 66, No. 11, pp. 1371-1386, November, 1978

[3] J.B. Brenner, "Conceptual Model for Integrity Control in Open Systems," ISO/TC97/SC16/N28, February, 1978

[4] Howard Anderson, "What is Electronic Mail?" Telecommunications, pp. 31-54, November, 1978

[5] T. Murawski, "Faxpak Store and Forward Facsimile Transmission Service," Electrical Communication, Vol. 54, No. 3, 1979

[6] G. Lapidus, "Facsimile Systems begin to link up with Computer Networks," Data Communications, January, 1976

[7] R.F. Bader, "Message Switching of Facsimile Data," Telecommunications, November, 1979

[8] J.C. Stoffel and C. Ramchandani, "Analysis and Design of a Store and Forward Facsimile Switching Node," ICCC Proceedings, pp. 81-87, August 1976

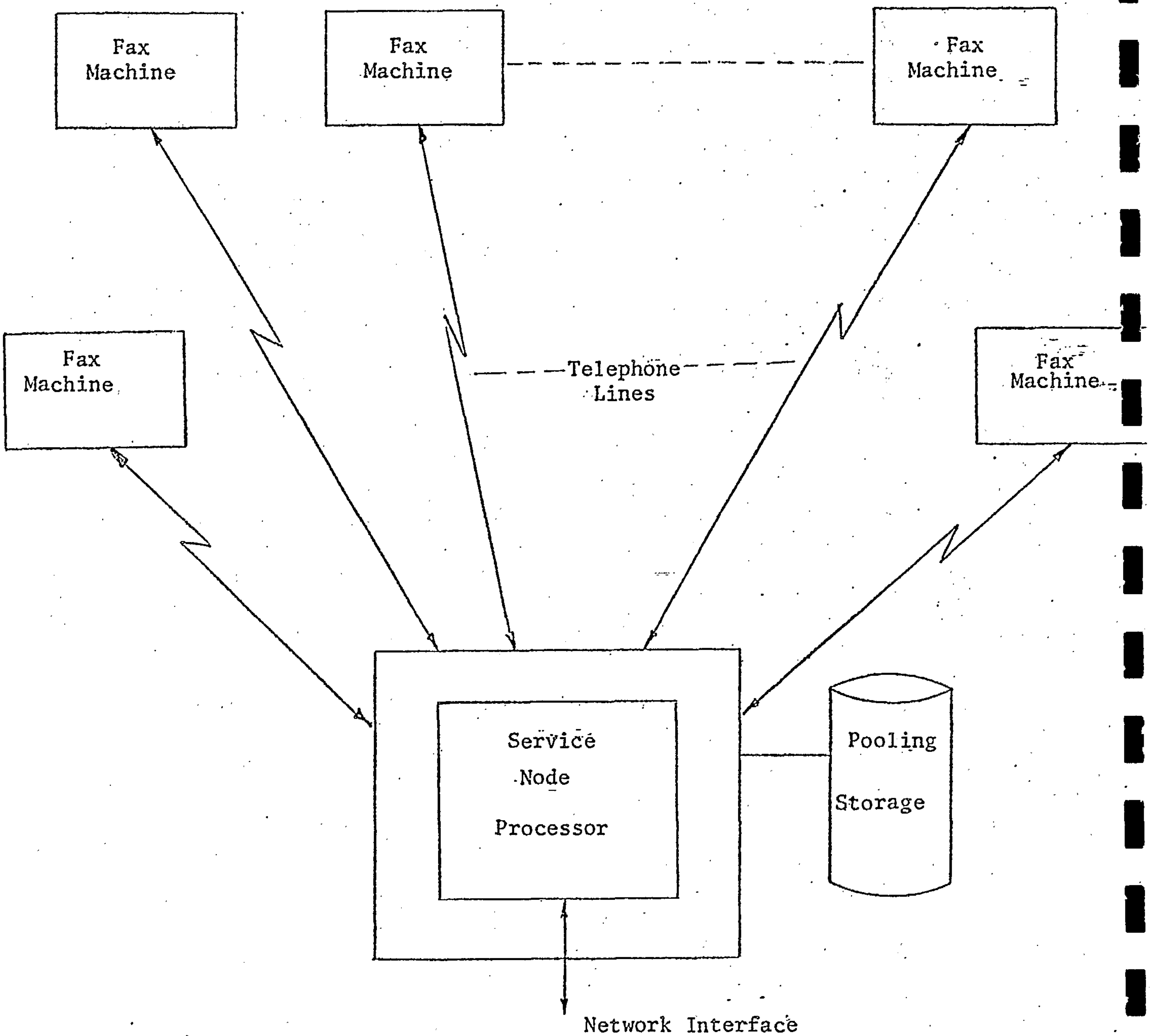


Figure 5.1: FACSIMILE SERVICE NODE CONFIGURATION

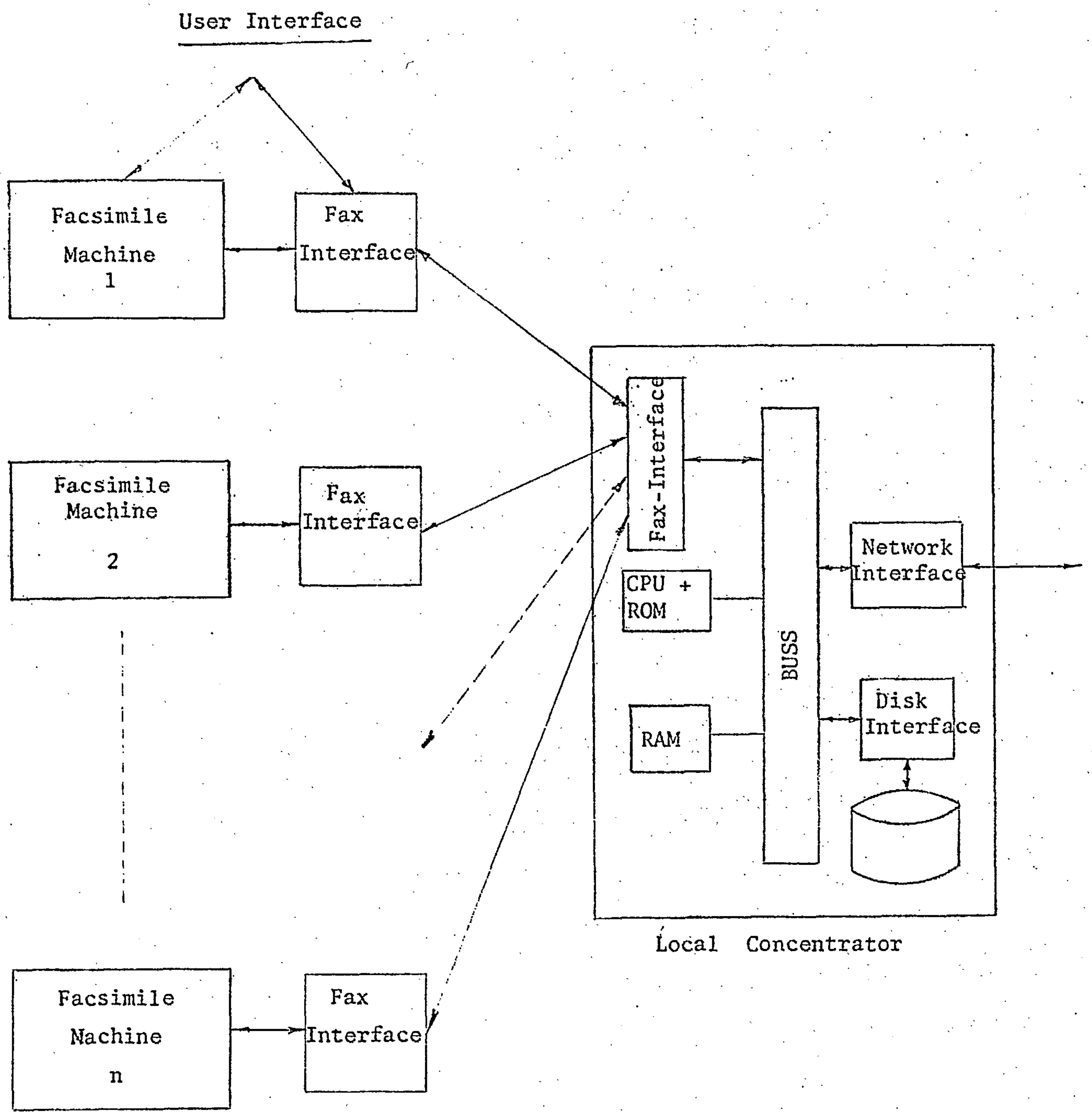


Figure 5.2: LOCAL CONCENTRATOR CONFIGURATION

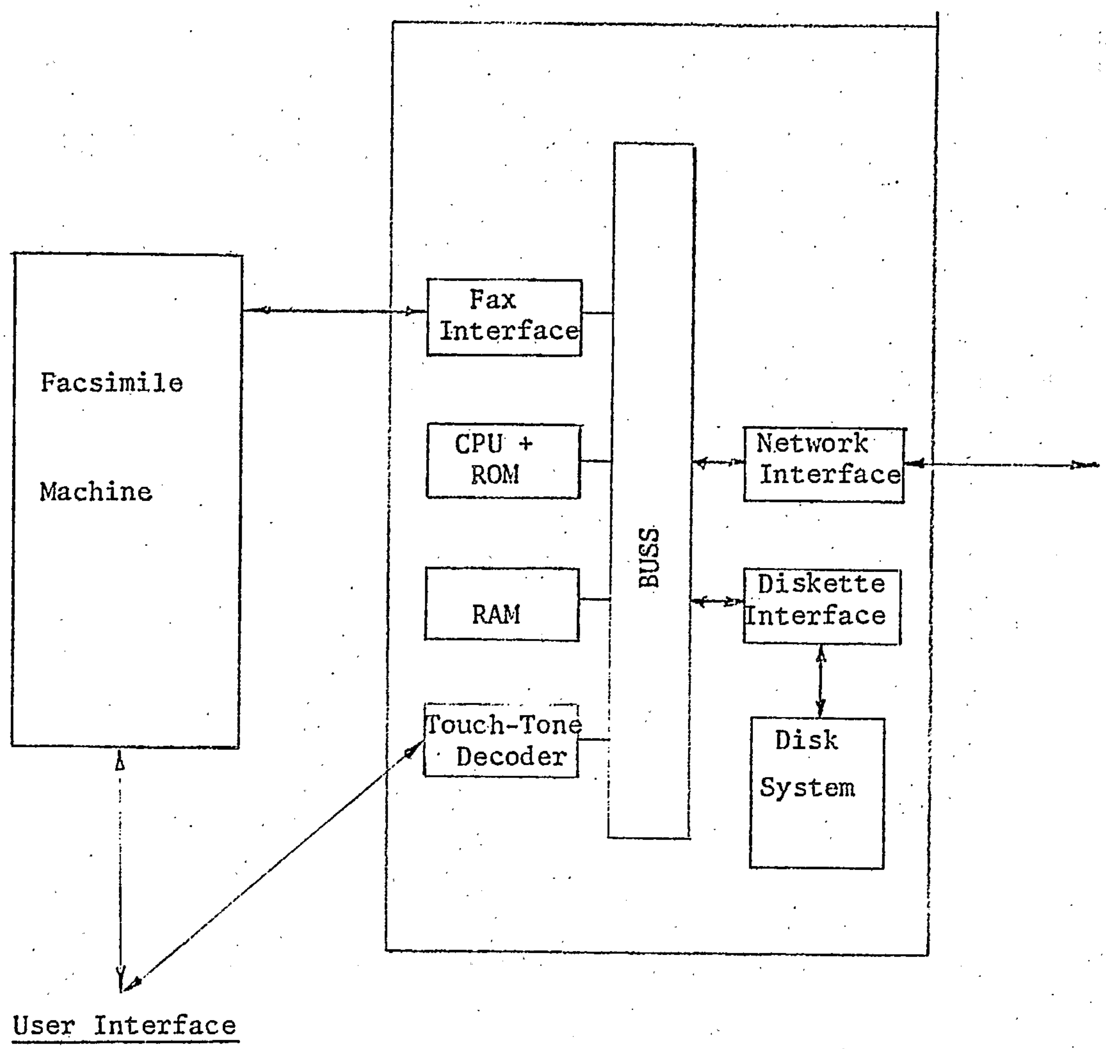


Figure 5.3: FACSIMILE FRONT-END CONFIGURATION

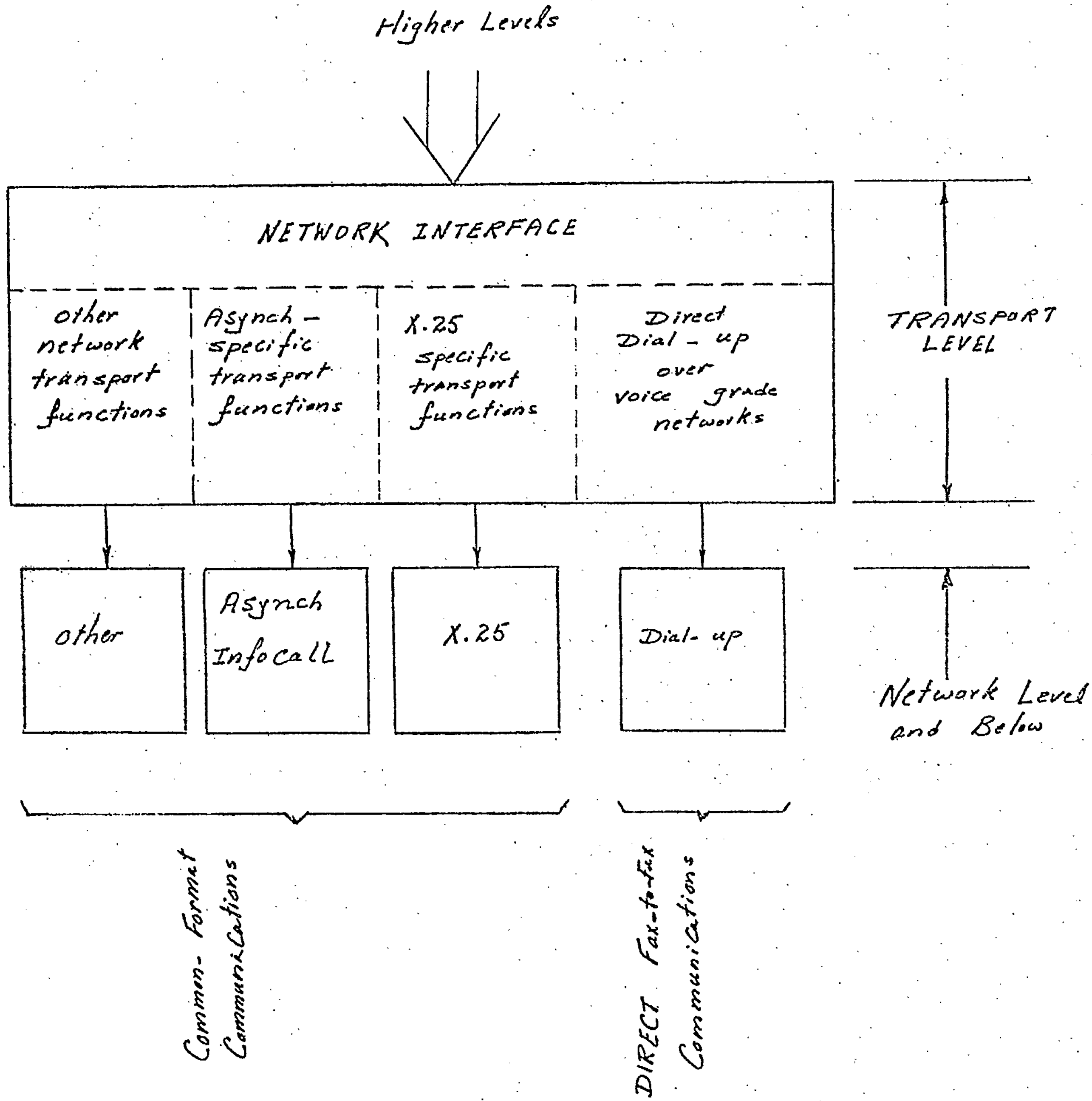


Figure 5.4: TRANSPORT LEVEL SERVICES IN FACSIMILE

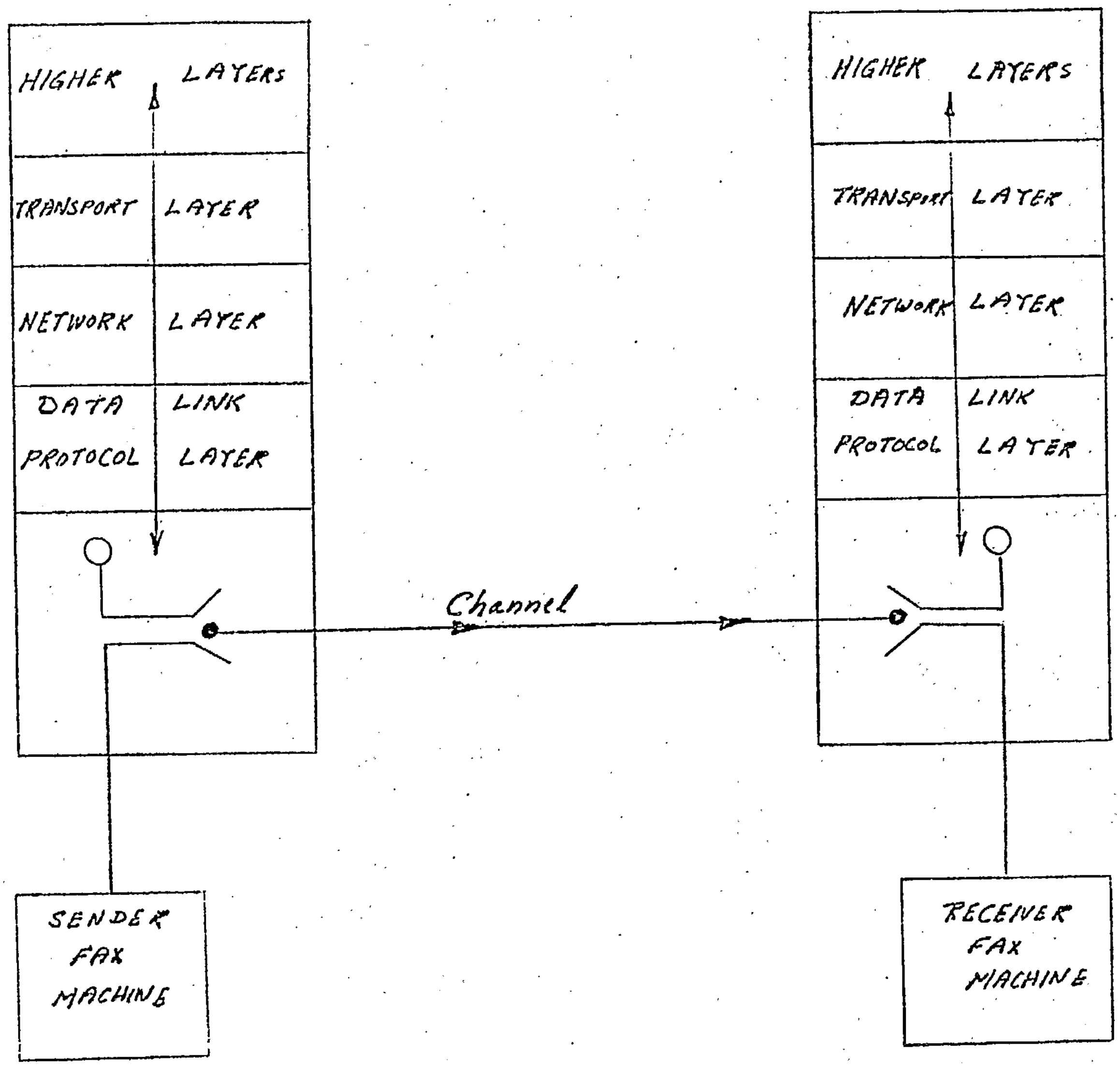


Figure 5.5: SWITCHING ROLE OF PHYSICAL LINK PROTOCOL IN FAX-TO-FAX COMMUNICATIONS OVER DIAL-UP LINES

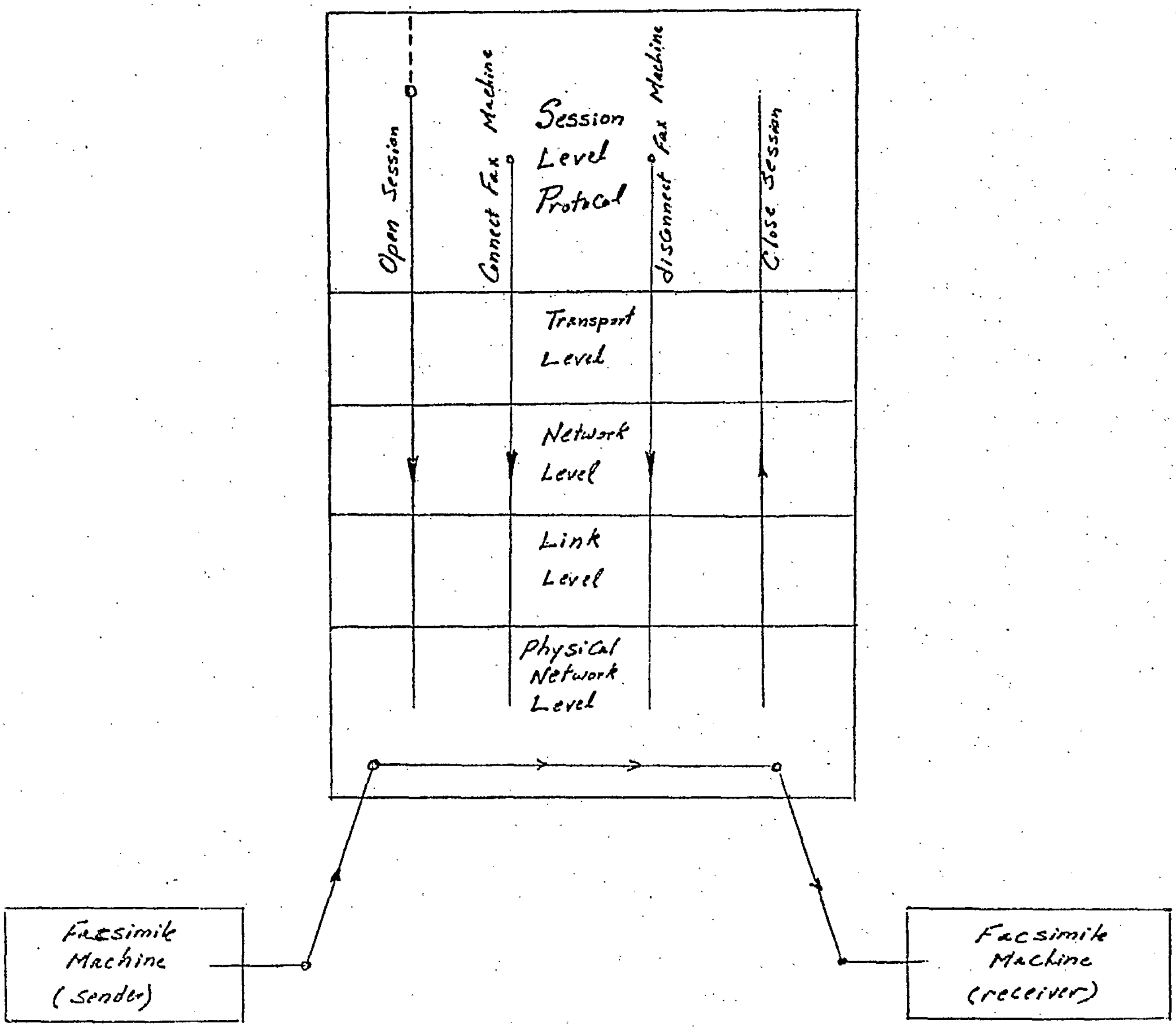


Figure 5.6: FUNCTION OF SESSION LEVEL PROTOCOL IN DIRECT FAX-TO-FAX COMMUNICATIONS

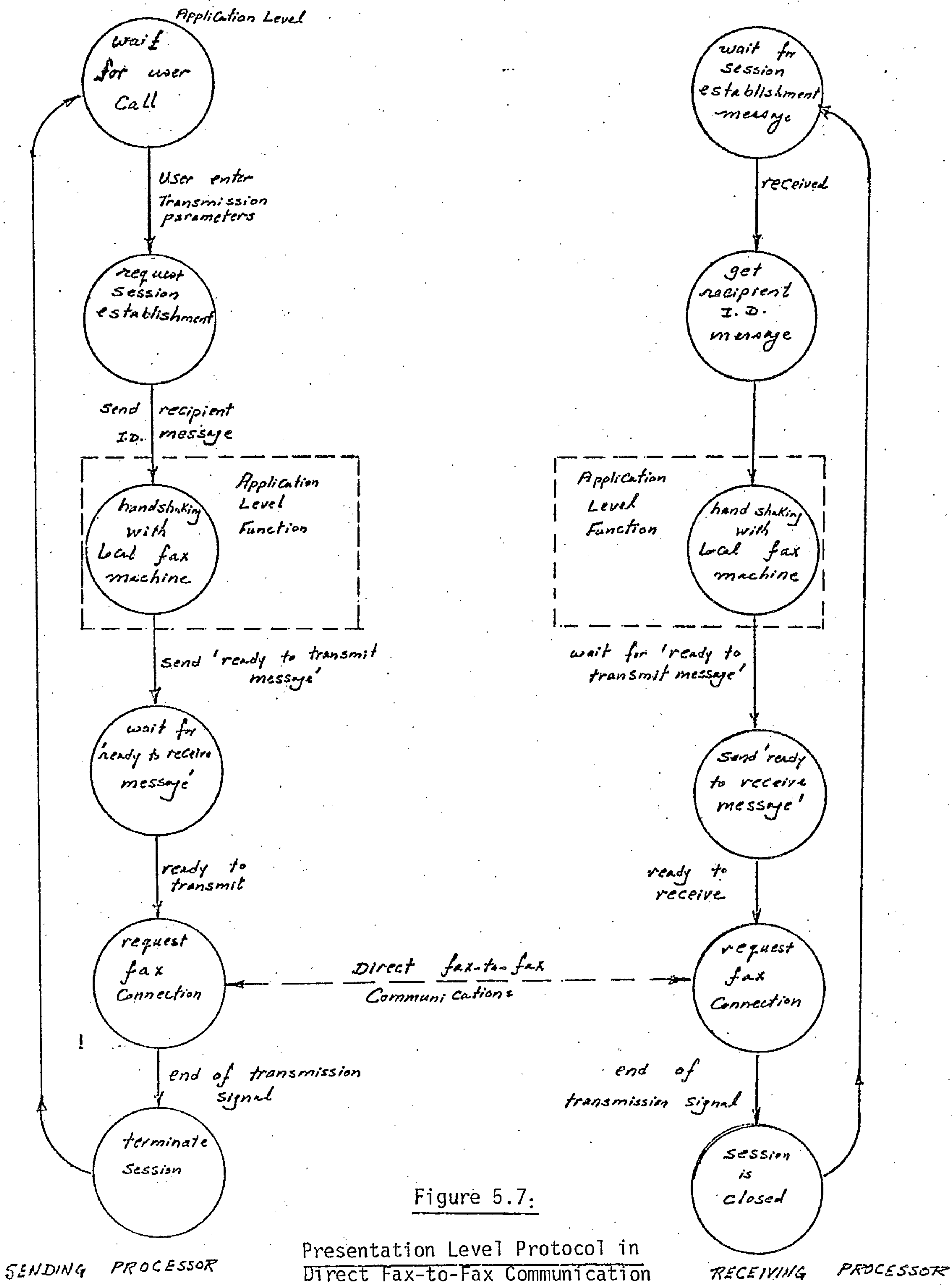


Figure 5.7:

Presentation Level Protocol in Direct Fax-to-Fax Communication

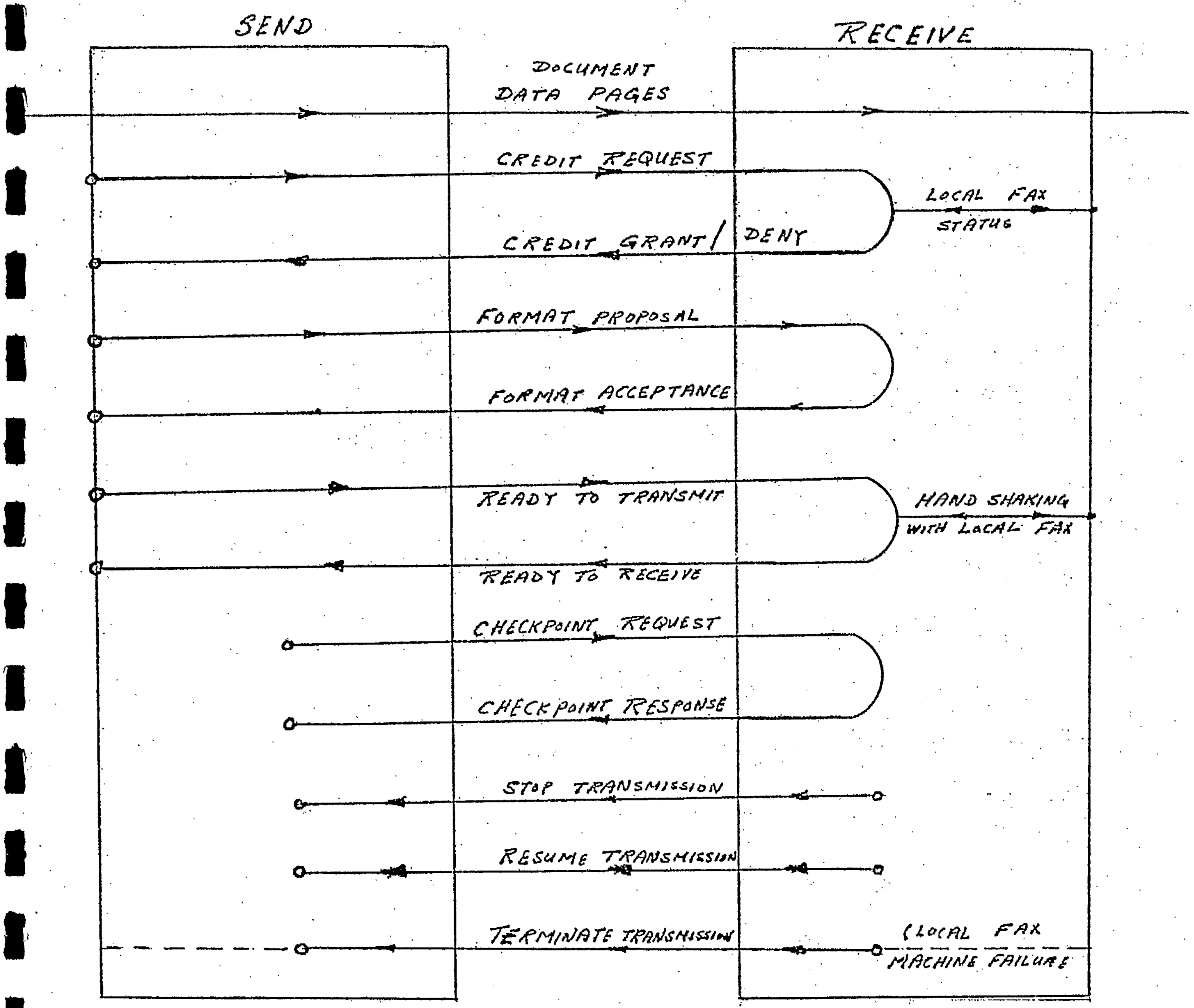


Figure 5.8: PRESENTATION LAYER MESSAGE FLOW IN FACSIMILE

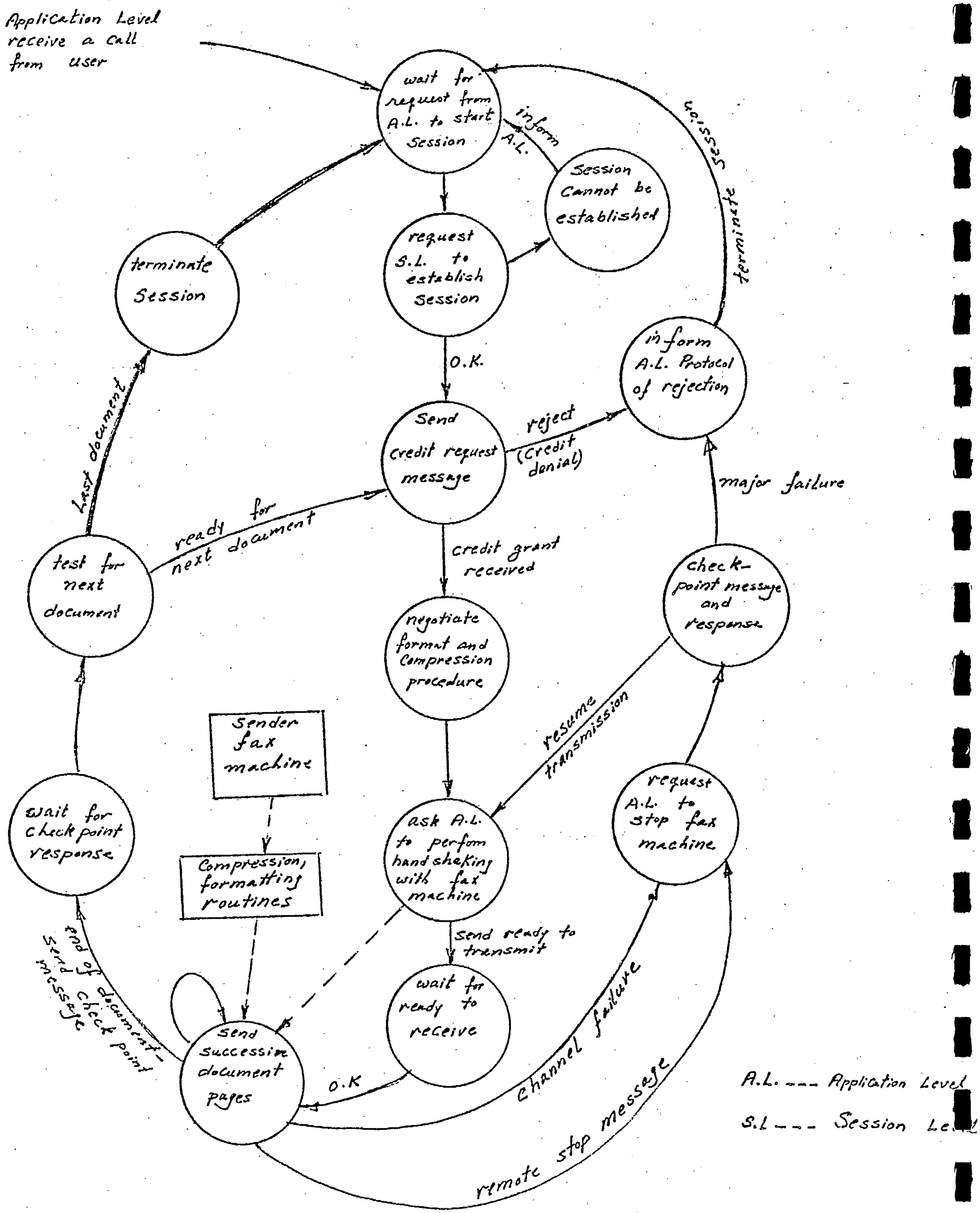


Figure 5.9: Presentation Level Protocol for General Fax Communications

Chapter 6

6. VIDEOTEX

6.1 Introduction to Videotex

The generic term VIDEOTEX has been adopted internationally as a facility which provides to the general public interactive retrieval of information. The Videotex systems currently in use or development typically enhance the characteristics of the home TV set such that the home user may connect to the Videotex service and search for relevant public, commercial or (possibly) private information. Some progress has been made in developing a standard description for Videotex Systems in the form of a CCITT draft recommendation [1,2].

In this chapter our intent is to examine whether and how Videotex systems may be described within the framework of the Open Systems Interconnection. Any issues which make such a description difficult, poorly defined or impossible to achieve, are discussed.

To accomplish these aims the chapter is organized as follows: Section 6.2 briefly describes some of the current Videotex systems in development and operation in Britain, France and Canada and outlines the potential uses of Videotex. Section 6.3 contains an overview of the elements of a Videotex system including the physical components and general operation. Following the CCITT recommendation [1,2], Section 6.4 outlines the functions and services provided by the Videotex architecture. Section 6.5 then presents a description of the Videotex system in terms of the layers in the OSI. Three alternative system configurations within the OSI framework are discussed in Section 6.6. Finally, a summary of the issues uncovered by the above exercise is presented in Section 6.7.

6.2 CURRENT VIDEOTEX SYSTEMS

The first systems which may be classified as Videotex were the British Cee-Fax, Oracle and Viewdata Systems. The Cee-Fax and Oracle systems, often referred to as Teletext systems, are one-way information transmission and retrieval systems which broadcast over existing television networks using spare lines in the video frame. These systems are not purely Videotex as they are not interactive systems. The British Viewdata system (now Prestel) is true Videotex in that it permits public users to interactively select from many pages of information stored in central data banks using the telephone network. These systems, and in fact all Videotex systems, use specially adapted television receivers to display requested pages of information. The British broadcast system transmits information by a so-called "synchronous" method in which each character position on the screen is tied to a specific position in the television line used to broadcast the information from the database.

The French systems currently under development use a non-synchronous protocol which is suited to a wider range of television formats than the British scheme. Their broadcast version is called Didon and their interactive (telephone) system is called Titan.

Both the British and French systems use a mosaic approach for displaying textual and non-textual (picture) information. Graphic images are constructed from specially identified coded graphic characters fitted together as individual pieces of a mosaic pattern.

The Canadian Department of Communications meanwhile is developing a Videotex system called Telidon [6,7,8,9] which intends to integrate both broadcast and interactive capabilities and which provides improved graph images independent of terminal design and communications media. Coding elements known as PDIs (Picture Description Instructions) are used to drive a micro-processor based terminal which interprets these codes as geometric structures and photographic streams. Free text is also supported.

Videotex services may also be extended into the following areas:

- terminal to terminal (subscriber to subscriber) communications and electronic mail
- transactional services (electronic funds transfer, booking systems and calculation facilities)
- direct access to large blocks of information for off-line perusal (catalogues, newspapers)
- downline loaded computer programs (computer games, special purpose packages)

6.3 THE ELEMENTS OF VIDEOTEX

Following the CCITT draft recommendation, the major features of Videotex systems can be described as follows:

6.3.1 Physical Components

Terminal - The terminal permits the user to interact with the Videotex service via a data entry device and a visual display (TV monitor). Three types of terminals are generally considered:

1. Numeric keypad: for public user access in the home environment where an inexpensive terminal is desired.

2. Alphanumeric keyboard: for commercial applications where cost is not a major factor.

3. Graphical input devices: for facilitating the entry of graphical information into the databases by the information providers.

We will restrict our attention to the keypad type as the others have not been considered in detail in the literature.

The terminal is typically intelligent (i.e., a microprocessor) so that it may buffer data, encode/decode transmitted and received information, perform terminal identification and handle the communications protocols.

Database - The database is a set of information which can be accessed by or forwarded to users of the Videotex service. A host computer stores one or more databases. Two types of host computers are defined:

1. An internal host computer operated by the provider of the Videotex service which provides mainly public information, and

2. An external host computer operated by commercial or government organizations which typically provide specialized information services.

   The database is usually stored in a tree structure with various levels of index. The upper levels of the tree form a database directory or access structure. The smallest unit of information is a frame and a page is a set of one or more frames.

Communications - Videotex services can be provided using both broadcast systems (cable, TV) and telephone networks. Because interactive systems may be possible over two-way cable systems, Videotex systems are classified by communications media as follows:

1. Broadcast Videotex - one-way cable or through the air
2. Interactive Videotex - Telephone network
3. Broadcast Interactive Videotex - two-way cable

6.3.2 General Operation

Display Modes - A frame of information for display may be composed of graphic elements which are interpreted by the terminal as text, symbols or pictures. Graphic elements may belong to one of the following categories:

1. Alphanumeric - various sets of alphanumeric graphic elements (text) may be selected.

2. Mosaic - blocks of various patterns which may be used to compose drawings.

3. Dynamically Redefinable Character Sets - specific elements defined by down-line loading the terminal with character defining information.

4. Geometric - drawings of points, lines, curves generated by terminal given characteristic parameter values.

5. Photographic - point plotting (facsimile-like). Other features like cursor control, motion, scrolling, colour, background/foreground highlighting and audio capability are either included or planned as part of the display features of the terminal.

6.3.3 Types of Connection

Terminal to Videotex Service - To retrieve information from host computers; retrieve billing and other management information; enter information into a database.

Videotex Service to Terminal - May initiate a call to a suitably equipped terminal in electronic mail applications.

Host to Videotex Service - Main functions would be for transfer billing and other management functions.

6.4 FUNCTIONS AND SERVICES IN VIDEOTEX

Following the CCITT recommendation, the following summarizes the proposed levels for Videotex (note that these do not correspond to the Open System levels)):

The Application Level - This is the highest level in which the user (via the terminal) interacts with other applications provided by the Videotex service.

The Service Level - At this level the user interacts with the Videotex service mainly to select the desired applications via the index (tree) structure. Terminal identification and usage logging is also performed at this level.

The transport Level - Provides the connection which links the terminal to the Videotex service.

The facilities and functions provided and performed by the Videotex service are listed below. Identification codes are introduced for each item to facilitate reference in the next section. Observe that a single code (R) has been used for the retrieval functions to save space. These are itemized in the CCITT draft recommendation [1]:

A: Access to Videotex and Host Services:
- A1: data connection to the services
- A2: connection to the Videotex service and host applications

D: Disconnection from Videotex:
- D1: leaving the Videotex (or Host) service
- D2: clearing the telephone or data connection

T: Terminal identification

U: Usage logging (for billing)

C: Control functions:
- C1: clear an unwanted entry
- C2: interrupt action in progress
- C3: terminate an entry as valid

S: Service functions
- S1: select an application provided by Videotex
- S2: return to point where first effective choice in Videotex service offered
- S3: leave Videotex service
- S4: leave Videotex service with billing information
- S5: provide billing information without leaving service

R: Retrieval functions: various functions which allow the user to search through the database.

P: Presentation services:

- P1: terminal capability determination (for further study)
- P2: character set selection (i.e., French, English, Latin, etc.)
- P3: display mode selection (alphanumeric, mosaic, geometric, photographic, etc.)
- P4: translation control (performance of the code set translations, picutre generations, etc.)

6.5 VIDEOTEX IN TERMS OF THE OPEN SYSTEM

In this section we consider the Videotex system as outlined in Section 6.4 within the framework of the Open Systems Architecture. We consider the Application, Presentation, Session and Transport (and below) layers in turn:

6.5.1 Application Layer

Our approach is to define three applications which will contain the application level facilities and functions described in Section 6.5:

1. Terminal Application
2. Videotex Service Application
3. Host Service Application(s)

We will now define these applications and identify the functions and facilities in each:

1. Terminal Application - The Terminal Application may be considered to consist of three parts:

   a) Keyboard Drivers: These routines buffer-up data from the keyboard; recognize local activation characters for clearing unwanted entries (C1); terminate a terminal entry and pass the buffered message to the Presentation Layer (C3); recognize an interrupt or attention sequence (C2) and pass it to the Presentation Layer (which in turn formats a session-interaction-unit for the session level).

   b) Display Drivers: The display drivers are application programs which accept display data from the Presentation Layer and perform the appropriate actions for outputting video to the display interface unit.

   c) Access and Control of the Videotex Service and Host Service Applications: The Terminal Application through local interaction with the user controls access to these applications. Control is initially passed to the Videotex Service Application (F1) and then to the appropriate Host Application (S1). Control is eventually returned to the Terminal Application using function S3.

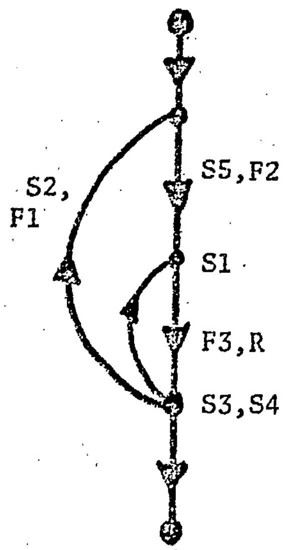
2. Videotex Service Application - The Videotex Service Application provides the following functions:

a) Service Assistance (F2)
b) Application Selection (S1 and S2)
c) Database retrieval (public information) (R)
d) Usage logging, billing transfer, billing storage and display
e) Billing transfer, storage and display (S4, S5)
f) Leaving Videotex service (i.e., returning to Terminal Application)

3. Host Service Application - The Host Service Application provides the following functions:

a) Host application assistance (F3)
b) Database retrieval (R)
c) Usage logging, billing transfer, billing storage and display

We can describe phases of application layer interaction using Open System Interconnection methods as follows:



Start Terminal Application: User turns on terminal, buffers initialized, application started

Videotex Service Phase: User interacts with Videotex Service for assistance and application selection.

Retrieval Phase: User interacts with Videotex Service for application assistance and information retrieval.

End Terminal Application: Terminal shut-down in orderly fashion

In Section 6.6 it will be shown that the Open System may be utilized in at least 3 different ways to achieve this application level phasing.

6.5.2 Presentation Layer

The presentation functions performed within Videotex can be associated with the Presentation Layer functions of OSI as follows:

1. Connection - The Terminal Application requests for connection to a Videotex Service or Host Service Application are passed through to the Session Layer (A2).

2. Presentation-Image Control - Negotiation of terminal options (P1), character sets (P2) and display modes (P3).

3. Data Transfer - Application level messages (i.e., user request messages or display frames) are translated according to the options selected in 2 above (P4).

4. Termination Phase - Disconnection using the session layer facilities (D2).

6.5.3 Session Layer

Certain Videotex functions can be associated with the Session Layer as follows:

1. Connection and disconnection of applications (which are tied one-to-one to corresponding presentation entities) - A2; D2.

2. Terminal identification - The connection request formulated by the Terminal Application (and forwarded by the Presentation Layer) contains the terminal identification since there is a one-to-one correspondence between the terminal user and the terminal application (T).

3. Data Exchange - Information frames would be carried by session-service-data-units. An interrupt detected by the Terminal application is forwarded (via the Presentation Layer) as an expedited session-interaction-unit.

6.5.4 Transport Layer and Below

Although the CCITT specification for Videotex implies simple telephone connections, it is possible to visualize the binding as a Transport Layer connection which interfaces the higher levels to a set of possible network services including connection-oriented, transaction-oriented and broadcast-oriented services. Therefore there is a direct mapping between the Transport Level in Videotex and the Transport Layer in OSI. Hence A1 and D1 functions are performed in this layer.

6.6 POSSIBLE ALTERNATE OSI CONFIGURATIONS FOR VIDEOTEX

The mapping of Videotex functions and services into the OSI framework suggest three possible configurations, each with their advantages and disadvantages. These configurations are depicted in Figure 6.6.1 and described below.

The three configurations consider a Terminal Application at one site, the Videotex Service Application at an internal host and a Host Service Application at some external host. In all cases it is assumed that a Virtual File Service is available to both the internal and external hosts which provide their database services. The Videotex Service and Host Service Applications contain the retrieval functions as we have already specified. We now consider the three configurations in turn.

6.6.1 Manual Directory

In this case the Videotex Service does not provide a first level directory as part of the interactive service. The user must use a form of "yellow pages" to determine the global title (i.e., address) of the internal or external host services that he desires to access. This approach simplifies the terminal application but places more responsibility for access in the hands of the user. Figure 6.6.1(a) illustrates a terminal session with the Host Service Application and another session with the Videotex Service applcation.

6.6.2 Local Directory

This second case implies that the terminal application can access a local directory which provides the terminal application with the desired application addresses (see Figure 6.6.1(b). Hence the user may select services from an on-line directory and does not have to consider details of addressing. In this instance, however, an updating problem must be resolved: such a local directory must be updated as new and modified services are provided. Also, local directory storage may increase terminal costs.

6.6.3 Remote Directory

The third case associates a directory with the Videotex Service Application (see Figure 6.6.1(c). In this case the Terminal Application is required to establish a session with the Videotex Service Application first to perform a user controlled directory search. This search returns the address to the terminal application which automatically establishes a session with the appropriate application - the internal or an external host. Figure 6.6(c) illustrates this configuration. The Terminal Application first establishes a session with the Videotex Service Application where the directory search is performed and the selected application address is returned to the Terminal Application. The Terminal Application in this example connects to an external host and establishes a session with the Host Service Application there. Alternately, if the desired application is already at the Internal Host, a new session is not needed and the Terminal Application allows immediate access to the retrieval functions in the Videotex Service Application. This approach implies greater utilization of communications and remote processing but avoids the update problems of the second case and the lack of on-line directory search of the first case. This case may provide other benefits associated with controlling user access to restricted applications.

The cases presented in this section are not intended to be exhaustive. Their purpose is only to demonstrate that the Videotex system can be modelled within the OSI architecture successfully.

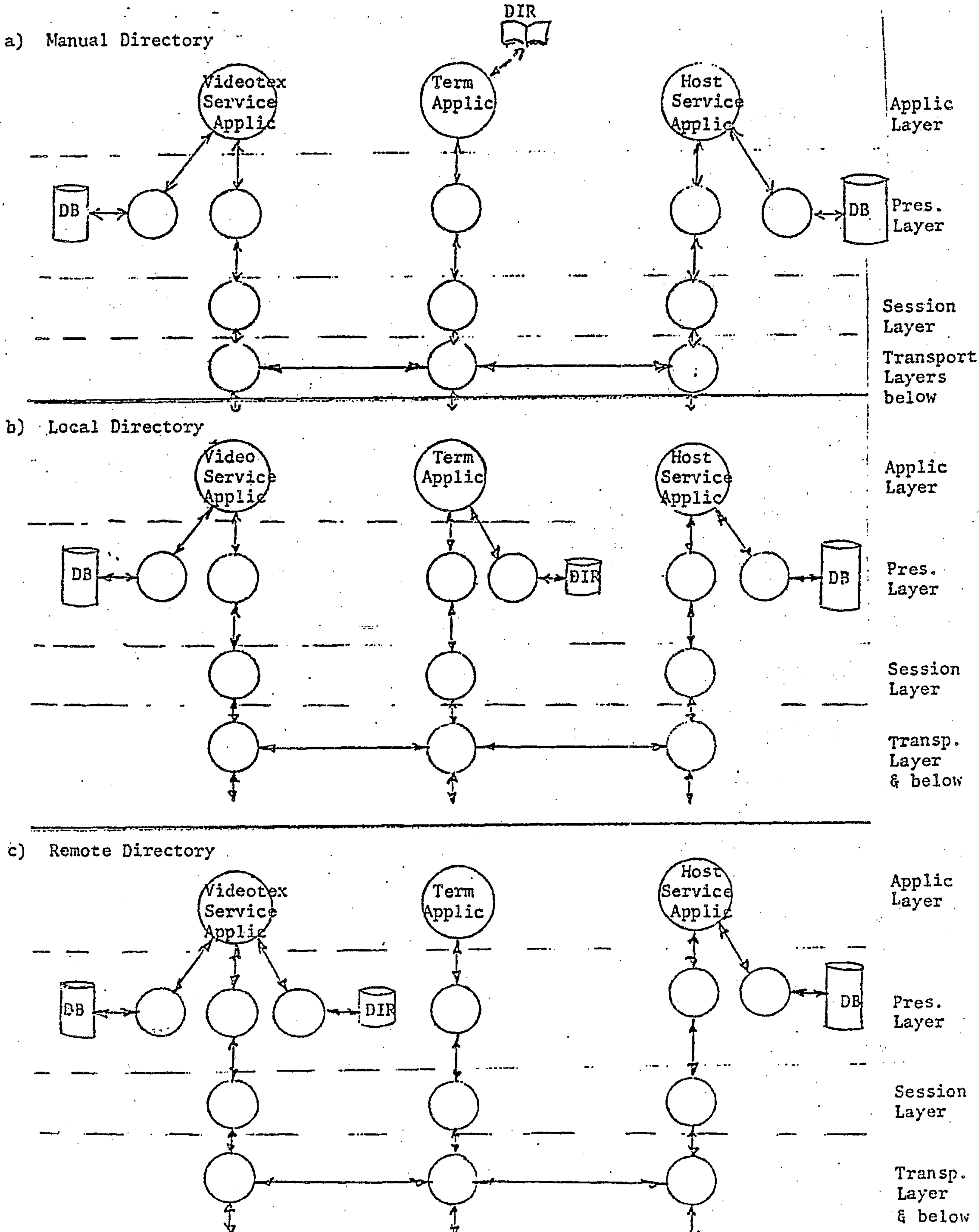


Figure 6.6.1: Alternate Videotex Configurations

6.7 ISSUES RELATED TO VIDEOTEX

In examining the issues concerning the application of the OSI to Videotex, we must first recognize that OSI is not completely defined in many areas and that the CCITT draft recommendation for Videotex is in its early stages of development. Therefore all issues raised may be quite constructive in the refinement of standards work in both arenas.

The primary benefit of the modelling exercise in Section 6.6 is that it demonstrates that the OSI can accommodate some of the candidate ways of conceptualizing Videotex systems. Some of the advantages and disadvantages of each approach were pointed out. All Videotex functions could be accommodated within the OSI.

6.7.1 Missing Functions

Several functions provided for in the OSI are not currently found in the Videotex recommendation. We now consider these in turn:

Presentation Layer:

- encryption - for public information encryption may not be needed; however, if Videotex is to be extended to commercial applications like electronic funds transfer, electronic mail, etc., this feature will be required.

- compression - although data compression is inherent in the geometric display mode it has not been specified for the text mode. Electronic mail and word processing applications of Videotex demand this capability.

Session Layer:

- blocking - because all retrieval is on a frame by frame basis, blocking is probably not needed. Extensions to other services may change this however.

- recovery - all recovery is currently the responsibility of the user. Session level recovery may be important however, since the user may not be happy to pay for retrieval services that are lost.

Transport Layer and Below:

- end to end sequencing and blocking - It is anticipated that these facilities are not needed for simple Videotex. More sophisticated extensions of Videotex, however, would require these facilities for recovery and efficiency reasons.

- flow control - All flow control is done by the user in a simple Videotex system. For the reaons outlined above flow control mechanisms may eventually be needed.

- optimization and class of requested service - Not really needed for the home user (see comments above).

6.7.2 General Issues

Transfer Billing: Transfer billing between various services has not been considered in any detail in the OSI or CCITT Videotex documents. The mechanisms for transferring billing between the communications providers, the host service providers and the Videotex service provider will be a complex area to address. The OSI suggests that should be incorporated on an application-management process in the application layer. This implies the appropriate billing "hooks" would be imbedded in the host service and videotex service applications to record the usage and special applications would support aggregation, transfer and reporting of customer usage for billing purposes.

Application Definition: In the OSI it is suggested that the virtual filestore is closely associated with the presentation layer. For the purposes of Videotex systems specification, it is imperative that this area be studied in further detail. We have suggested that database applications be specified at the application level in each host. Terminal applications would invoke connections to these database applications via the lower layers for information retrieval. The presentation layers at both ends would invoke the appropriate virtual file protocol which would standardize access to database applications with different access languages and formats.

Interactive-Broadcast Services: How broadcast services such as (one-way and two-way) and over-the-air TV fit into the OSI layers has yet to be considered in detail.

To illustrate the issues here that require careful examination we consider the typical Interactive Videotex service shown in Figure 7.2(a). The diagram illustrates the connection of a terminal application A_t to a host service (database) application A_h via the telephone network and presentation and session entities in the terminal and the host. In the case of the telephone network, transport connections are manually invoked by the user and the transport and below layers are not used. The presentation entities perform the virtual terminal protocols required to manage the PDI codes. The terminal application drives the TV display electronics which produce the user images.

In Figure 7.2(b) we present a natural Interactive-Broadcast Videotex configuration. In this instance the terminal application establishes a connection with two presentation entities P_t and P'_t. As in the case above, P_t invokes a two-way alternate session with the host application on behalf of A_t via S_t and S_h. Meanwhile, a permanent one-way connection exists between P_t and P'_t via a one-way cable TV network. In this case the telephone system supports keyboard commands issued by the user and any return control messages (i.e., turn-around and acknowledgement messages); the cable network supports the PDI display traffic in the return direction only. The S'_t to S'_h connection is permanent because all videotex terminals and hosts are physically connected to the cable network at all times. Hence the session level functions of session establishment and termination are not used for S'_t and S'_h and these functions need only carry out session-level addressing duties (i.e., S'_t selects session-data-units from S_h and ignores all others. A desirable facility would be to allow group addressing at the session level as this would permit sending the same PDI frame to several users simultaneously).

Another configuration that can be considered supports the two-way keyboard traffic via a packet-switched network and the one-way display traffic through a cable-TV network. Figures 7.2(c) and (d) illustrate two approaches for describing such a configuration under OSI. The first approach switches the two data streams at the application level while the second approach performs this switching at the transport level. Although the second approach fits into OSI more naturally, the first appears to lend itself to simpler implementation (note: N_t, N_t', N_h, N_h' are network layer entities and T_t, TI_t, T_h, T'_h are transport layer entities).

These configurations illustrate that more than one approach may be possible and that therefore many advantages and disadvantages for each alternative will arise. Further, as OSI has not yet formally defined broadcast systems within its architecture, many issues regarding addressing, multi-plexing, flow-control and recovery are yet to be considered in detail. We may therefore expect that the solutions to problems encountered in broadcast Videotex systems will be of immediate importance to the development of the OSI standard.

6.8 References for Chapter 6

1. CCITT, "Meeting Report", COM I-No. 199-E, Study Group I's Videotex Working Party, 2 November 1979.

2. CCITT, "Draft Recommendation S", Study Group VIII, Videotex Working Party Meeting, Paris 17-19 November 1979.

3. CCITT, Draft New Question VII/xx, Study Group VII, Com VII - No. -E, September 1979.

4. CCITT, "Proposed Network-Independent Transport Service Interface for Telex", Study Group VIII, January 1980.

5. CCITT, "Reference Model for PDN Applications", Study Group VII.

6. H.G. Bown, C.D. O'Brien, D.F. Parkhill, W. Sawchuk and J.R. Storey, "Telidon: A Public Access Information System", Engineering Journal of the Engineering Institute of Canada, April 1979.

7. H.G. Bown, C.D. O'Brien, W. Sawchuk and J. Storey, "Telidon: A New Approach to Videotex System Design", IEEE Transactions on Consumer Electronics, Vol. CE-25, No. 3, July 1979.

8. J.C. Smirle and H. Bown, "New Systems Concepts and Their Implications for the User", May 1979.

9. H.G. Bown, C.D. O'Brien, W. Sawchuk and J.R. Storey, "Picture Description Instructions for the Telidon Videotex System", Department of Communications Internal Report, November 1979.

10. G.V. Bochmann, "Possible Database Structures for Future Applications of Videotex Terminals", University of Montreal Contract to Report to the Department of Communications, May 1979.

Chapter 7

7. Summary of Issues

7.1 Introduction

Chapters 3 through 6 have discussed Open System issues relating to particular application areas. In this chapter the key issues are summarized and discussed in the broader context of the range of application areas studied in this report.

Section 7.2 summarizes general issues.

Section 7.3 summarizes issues particular to two application areas with special characteristics and problems: Facsimile and Videotex.

Finally, section 7.4 summarizes issues associated with multi-function terminals.

7.2 General Issues

7.2.1 Introduction

General issues, which are discussed in more detail below, are as follows:

- application-dependence of levels;
- network-dependence of levels;
- inter-dependence of levels;
- duplication of functions in various levels;
- lack of clear definition of the purpose of certain levels.

7.2.2 Application-Dependence of Levels

A desirable goal in the opinion of the authors is to confine application-dependent functions to the application level to the maximum extent possible. Experience has shown, however, that the application areas tend to have an influence on the structure of all levels, from the top down. Specific issues are discussed below.

a) The "Virtual Application" approach to the application and presentation levels may generate undesirable application-dependence on the presentation level, as discussed in Chapter 3, Section 3.7.4.

b) Different applications may generate a need for different types of level services and functions, as discussed for the Session Level in Chapter 3.

c) Certain applications may have restricted needs for functionality of a level, which may make it impossible to share the level in a multi-function terminal. An example is the support of concurrent sessions which is proposed in Chapter 3 but which is apparently not supported by the Teletex session level as discussed in Chapter 4.

d) Other applications may require the physical allocation at a network node of the different functions to separate pieces of equipment in a manner which blurs the distinction between open system levels. An example is facsimile, as discussed in Chapter 5.

e) Certain applications may not easily provide for the presence of all levels in all network nodes, thus making it difficult to fit these applications into the Open System Framework. An example is the Videotex user terminal as discussed in Chapter 6.

7.2.3 Network-Dependence of Levels

Just as the application area tends to have an influence on the structure of all levels from the top-down, so the network service (or services) planned for use with a particular application tends to have an influence on all levels from the bottom-up. Specific issues are discussed in more detail below.

a) Caller Identification: Lack of caller identification in certain network services may require higher levels to perform this service, as discussed for X.21-like services in Chapter 3.

b) Broadcasting in the Open System Framework: Broadcasting is not defined at any level, but its effect could be felt through all levels. Broadcasting can be handled by separate sessions or by special broadcast sessions. Separate sessions are very inefficient particularly if handled sequentially, but fit easily within the existing Open System structure. The session level may have to control broadcasting by setting up many sessions as requested by higher levels. Special broadcast sessions are attractive when the network provides broadcast services. Then the transport level could select the network service for broadcasting at the request of the session level.

There is an obvious possibility of multiple transport connections for one session. How would these be handled? What about reverse traffic acknowledgements for a broadcast service? For example, as discussed in Chapter 6, Videotex cannot have such acknowledgements because the networks are one way.

c) Access versus Routing Functions in the Network Layer

The concept of the network layer performing routing functions to select appropriate routes between network addresses appears inconsistent with existing public network services such as X.25. What appears to be more appropriate for such networks is for the network layer to select the type of service which may affect routing (e.g. a priority service). However, participation in routing is explicit for networks like ARPANET. The role of the network layer in routing is thus quite unclear. Specifically, the allocation of access and routing functions between different levels is unclear.

d) Use of Multiple Networks in One Session

The possibility of using multiple networks for one session is a distinct possibility which may produce difficulties. As discussed in Chapter 6 and in item (b) above, the Videotex service may use two networks, including a one-way broadcast network. A single user-session involves the use of both networks. The implication for the session level are unclear.

7.2.4 Inter-dependence of Levels

It is a desirable goal to make each level as independent as possible of all the others. Internal changes in one level should not affect other levels. The following issues may have an effect on this goal.

a) The "virtual application" concept implicit in the ISO Model requires that functions fundamental to the nature of various application areas must be distributed through several levels;

b) The selection of network services required because of the application and network-dependence of levels may require that network level sercice parameters be passed from all higher levels, including the application level.

c) The issues raised in 7.2.2 and 7.2.3 affect inter-dependence.

7.2.5 Duplication of Functions in Various Levels

There is a tendency to define certain levels such that functions performed in other levels are duplicated. This covenent particularly applies to the following functions:

a) Flow control at the presentation level may duplicate the function of flow control in the lower levels as discussed in Chapter 4 for Teletex. Flow control at the session level may duplicate transfer control functions better performed at higher levels and flow control functions better performed at lower levels as discussed in Chapter 3, Section 3.7.3.

b) Recovery of data failures at the session level duplicates functions better performed at higher levels, as discussed in Chapter 3, Section 3.7.3.

c) Identification of calling terminal ID at the session or transport level duplicates functions provided at the network level by some network services.

7.2.6 Lack of Clear Definition of Purpose of Certain Levels

The lack of clear definition of the transport level is a particular problem. It is clear that its functions could be allocated to other levels.

7.3 Particular Issues Associated With Facsimile and Videotex Application

7.3.1 Facsimile

In addition to the general issues discussed in Section 7.2, several particular issues associated with the facsimile application have been identified. In this section, we summarize these particular issues for each level of the Open System Interconnection protocols in the facsimile application.

7.3.1.1 Issues at Transport and Lower Levels

(a) In the direct Fax-to-Fax communications mode, the transport level is expected to set up connections with three phases: Establishment Phase, Data Transfer Phase and Termination Phase. The first is initiated based on a request from the session level. The second phase involves switching the communications channel (at the physical link level) to allow the two Facsimile machines to communicate directly. The practical feasibility of this phase is still open to question. Also the execution of the termination phase requires signalling from the physical link level; an aspect which requires further scrutiny. Failure to detect the termination tones may leave the connection open for an indefinite period of time.

(b) The choice of which service mode to use (common format or direct Fax-to-Fax communications) has to be decided at higher levels since it is dependent on the types of Fax machines used at each end. Thus the higher level protocols are not network independent in the fullest sense. The alternative would be to leave this decision within the Transport Level, which would tend to complicate its structure and require passing several parameter values from the session level to the transport level.

Error discovery depends to a large extent on the adequacy of the interface to the facsimile machines and the completeness of the status signal set of these machines. In the direct Fax-to-Fax mode, the Transport Level does not control data transmission and will thus be unable to check service quality and possible errors. It is essential to examine the set of status signals of each facsimile machine to ensure that the Transport Level is constantly aware of the status of transmission and can inform the higher level protocols of any disorder.

7.3.1.2 Issues at Session Level

(a) Two different types of sessions are required for the facsimile application: (1) Sessions involving direct facsimile machines communications, and (2) Sessions in which facsimile data is transmitted in the form of data messages between higher level protocols. The decision regarding which session type to select depends on the type and compatibility of the communicating machines. Since this information is available at the application level, it seems that the decision will have to be made at this higher level. This naturally violates the concept of independence of the session level from the levels above it.

(b) In the direct Fax-to-Fax mode of operation, several functions executed at the session level, such as session reset and session close, are triggered by signals received from the facsimile machines. These signals are received at the physical link level and must be propagated to the layer above until they reach the session level. This means that certain mechanisms must be provided to ensure that the status of the channel is monitored continuously. The absence of such mechanisms can lead to some sessions being opened indefinitely while no data is being transmitted between the end nodes.

7.3.1.3 Issues at Presentation Level

(a) The large volume of data generated in facsimile requires vast secondary storage, especially in situations involving a large speed difference between sender and recipient fax machines. In this case, it is essential for the host processor to be able to interrupt the sending machine for any duration of time, followed by resumption of operations. The interrupt capability may be also needed in case of temporary failure of the communications link. The interrupt capability, however, is not a feature available on every fax machine (see Table 5.1).

(b) The broadcasting of fax data to multiple recipient sites can be accomplished by setting up concurrent sessions between the sender and the recipient sites. The presentation level protocol can then be executed simultaneously with each site. However, if any of the recipient machines is much slower than the sender, it will be the responsibility of the recipient machine to provide enough space to buffer the data which is arriving at high speed but being delivered to the recipient at slower speed. The recipient host takes this difference into account at the time of exchanging the "credit request" and "credit grant" messages.

(c) One issue that must be resolved in the broadcasting to multiple sites is the action that must be taken by the sender host after it receives a "stop transmission" message from a recipient site which is facing a storage overflow problem.

7.3.1.4 Issues at Application Level

(a) The addressing information in facsimile which is used to identify the recipient to the Open Interconnection Protocol has to be submitted separately, i.e., not through the fax machine, to the system. This limitation is due to the fact that facsimile data is not analysable by the digital processor. However, it gives the user interface a special significance in the sense that two separate actions have to be performed: getting the fax machine ready to start document transmission, and entering the addressing information through a different interface to the digital processor. Special attention must be given to the requirement of document broadcasting to multiple locations, particularly in Configurations I and II, to define how multiple recipient addresses can be entered.

(b) Addressing data values (station and substation numbers) must be standardized to a universally acceptable format. As well, there is a need for defining the format of the first page of the document which contains the information required for sorting and delivery of facsimile documents.

(c) Several delivery aspects remain to be defined: procedures for handling rejected documents, handling requests for retransmission of documents received with unacceptably low quality, and handling registered documents, i.e., documents for which acknowledgement of receipt is expected.

7.3.2 Videotex

7.3.2.1 OSI Functions Not Defined For Videotex

Several functions provided in OSI are not currently defined for Videotex:

a) Encryption: Encryption may be required if Videotex services are provided for commercial applications like electronic funds transfer, electronic mail etc.

b) Compression: Data compression would be required if Videotex is used for electronic mail and wordprocessing.

c) Blocking: Blocking information streams is not generally needed for current Videotex but other file and document oriented systems will require blocking as well as flow control.

d) Recovery: Error checking and recovery at the session layer may be required.

e) End-to-end sequencing and blocking: Although not needed at the transport level for ordinary Videotex Flow-control end-to-end sequencing and blocking will be needed for document based extensions of Videotex.

f) Optimization and class of requested service: These services are not needed for home user applications; however, commercial users may benefit from such capabilities.

7.3.2.2 General Issues

The following important issues require extensive study:

(a) Transfer Billing: The mechanisms for transfer billing of customer usage between the communications providers, the host service providers and the Videotex service provider will be a complex technical as well as administrative area to address.

(b) Application Definition: It is proposed that database (or filestore) applications be carefully defined within the application layer of OSI with corresponding presentation layer services specified. Specifying database applications along these lines will greatly simplify the modelling of Videotex within the Open System.

(c) Interactive-Broadcast Services: The detailed implications of broadcast services within both OSI and the Videotex standard have not been considered as of yet. Many issues related to addressing, multi-plexing, flow-control, recovery will have to be studied in depth.

7.4 Multifunction Terminals

The foregoing issues affect the viability of multifunction terminals.

If the levels are application-dependent, then many versions of each level will be necessary in a multifunction terminal, one for each application. Or complex parameterizations of each level will be required.

The same holds true if the levels are network-dependent.

The combination of application-dependence and network-dependence will greatly complicate the problem of choosing suitable combinations of special types of levels for particular multi-function terminals. In general, the problem is one of inter-dependence of levels.

To minimize proliferation of special versions of levels, there will probably need to be some duplication of function between levels.

Finally, there are special problems with including Facsimile and Videotex services in a multi-function terminal because of the difficulty in some cases of fitting them into the Open-Systems framework.

Chapter 8

8. Conclusions and Recommendations

8.1 Conclusions

This report contributes to the Canadian government's efforts to gain insight into the problem of developing unified multipurpose terminals capable of interworking over a variety of national and international network services. It contributes by identifying issues and problems associated with a number of practical applications. In terms of the specific objectives of the study, summarized in Section 1.2, the following progress has been made:

1. The practicability of the ISO/CCITT concepts and ideas for a layered model of open system interconnection relative to a number of practical applications has been examined in some detail. And a number of issues affecting practicality have been raised.

2. Problems with fitting the applications within the architecture without disturbing the latter's logical structure have been found and documented. In some cases these problems result simply from lack of clarity of interpretation of the architecture. But in a few cases there appear to be problems with the architecture itself. These have been documented.

3. Inferences have been drawn about the logical soundness of the architecture concepts and ideas. Particular attention has been paid to criteria for partitioning functions between layers. Some new criteria have been proposed and some problems have been identified.

With respect to the important sub-objective of defining as precisely as possible the requirements of the applications, this report provides diagrams following a consistent methodology which show services, message flows and, in some cases, basic protocols for the levels of the different applications.

8.2 Recommendations

It seems clear that considerable further work will be required before Open System standards emerge suitable for the range of applications discussed in this report. Some particular areas requiring further investigation are summarized below:

1. Transport level services and functions should be examined in more detail.

2. Criteria for partitioning of functions between the application and presentation levels should be established.

3. The requirements of emerging new application areas such as TRADEX, Electronic Mail and Communicating Word Processors should be formalized in the Open System context.

4. The many unanswered questions raised with respect to Teletex, Videotex and Facsimile services should be investigated further.

5. A specific study of multifunction terminal requirements should be performed, with a view to defining associated requirements for shareable levels of the Open System Interconnection model.