ECMA
EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-72

TRANSPORT PROTOCOL

January 1981

Free copies of this document are available from ECMA,
European Computer Manufacturers Association
114 Rue du Rhône – 1204 Geneva (Switzerland)

ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-72

TRANSPORT PROTOCOL

January 1981

BRIEF HISTORY

Work on a ISO Reference Model for Interconnection started in 1977 with the formation of ISO/TC97/SC16 and the development of a Reference Model, now ISO DP 7498.

In ECMA work on a Transport Protocol to be designed in accordance with Layer 4 of the ISO Reference Model started in 1978 with the formation of TG E in TC 9 (further TC 24, Communication Protocols) under the guidance of TC 23 (Open Systems Interconnection).

A first working paper on Transport Protocol was prepared in 1979 and a first draft later in the same year.

Cooperation with ISO/TC97/SC16 and CCITT was maintained during the development of the Standard, and Class 0 was introduced on CCITT request.
Work is now continuing on definition of Class 4.

This Standard has been passed as Standard ECMA-72 by the General Assembly of ECMA of Dec. 18, 1980.

TABLE OF CONTENTS

Table of Contents (cont'd)

Table of Contents (cont'd)

1. GENERAL

1.1 Scope

This Standard relates to the Transport layer entities of Open Systems conforming to the ISO Reference Model for Open Systems Interconnection. It provides:

- A definition of the set of Abstract Service Primitives through which the Services of the Transport Layer are made visible to higher layer entities.
- A specification of the protocol to be implemented by the end system peer transport entities comprising the Transport Layer and existing to provide the Transport Service.

The provisions of the Transport Layer Protocol are prescribed by both the definition of the Transport Service as given in this document and by the definitions of the various Network Services existing to provide interconnection of end systems.

Two classes of Transport Service have been identified, Connection oriented and Transaction oriented. This edition of the Standard addresses only the requirements for Connection oriented Transport Services based on the assumption that the underlying Network Service is either connection oriented or of the datagram type.

To meet the requirement for a reliable connection oriented service over a datagram network requires the definition of a sophisticated and robust Error Detection and Recovery Transport Protocol. It is intended that this should be met by the Class 4 protocol to be included in a future edition.

It has been recognized that the proposed Class 4 protocol is not restricted to use over datagram networks. It could also serve over unreliable connection oriented networks subject to a high incidence of unreported errors. However, in such a case not all the provisions of the proposed Class 4 may be needed. The possibility of a reduced Class 4 is therefore also envisaged.

1.2 Introduction

A general purpose Transport Protocol must satisfy requirements that vary with respect to both the quality of the underlying network services and the nature of the user application. Where the underlying network service meets the requirements of the user application, little or no enhancement is required from the Transport layer. Where disparity exists between the requirements of the user application and the quality of the underlying service, the Transport layer must provide the necessary enhancement. To meet these differing requirements the ECMA Transport Protocol has been designed in a modular way using the concepts of classes and options. Five classes have been defined ranging from Class 0 as required for simple terminals (e.g. Teletex), to Class 4 as required for, typically, datagram networks.

No difference exists between Class 0 and the Transport Protocol adopted by CCITT for the Teletex service.

The present edition of the Standard does not in fact include a definition of Class 4 procedures as required for datagram networks and parallel network connections exhibiting datagram like properties. These procedures are currently the subject of study both in ECMA and other standardization bodies. It is the intention of ECMA to ensure the maximum possible degree of compatibility between its own Class 4 procedures and those being developed elsewhere. For this reason ECMA work on Class 4 procedures has been deleted from this edition. Publication of the Standard in this form is being undertaken in recognition of an urgent user need for a Transport Standard covering the majority of currently perceived user applications.

2. CONFORMANCE

To conform to this Standard all the provisions of the specified class shall be implemented.

3. REFERENCES

ISO...* : Basic Reference Model of Open Systems Interconnectio
CCITT Recommendation X.25
CCITT Recommendation X.21
CCITT Recommendation S.70

4. DEFINITIONS

For the purpose of this Standard, the following definitions apply.

4.1 Transport Address

The means by which session entities can communicate using the transport service, and by which they are known to the Transport layer.

4.2 Transport Connection

A two way simultaneous data path between a pair of transport addresses.

4.3 Transport Connection End Point (TCEP)

A transport connection provided by the Transport layer, as seen by its user.

4.4 Transport Connection End Point Identifier (TCEP-ID)

The means for session entity to distinguish between different transport connections. It is provided by transport.

4.5 Transport Service Data Unit (TSDU)

A piece of data for the Session Control layer transferred by the Transport layer. Its size is architecturally unlimited but it has a distinct beginning and end.

* In preparation. At present ISO DP 7498

4.6 Transport Interface Data Unit (TIDU)

The amount of data transferred across the transport/session interface in a single interaction. The size of a TIDU is not necessarily the same at each end of a connection.

4.7 Transport Protocol Data Unit (TPDU)

A unit of data containing control information and (possibly) user data.

4.8 Network Connection

A point-to-point connection between two transport entities. A network connection can be permanent or established for a limited amount of time.

4.9 Network Service Data Unit (NSDU)

A piece of data transferred over a network connection by the Network layer for the Transport layer.

4.10 Network Address

The means by which a transport entity is known to the Network layer and by which it can communicate through the network service.

4.11 Expedited Transport Protocol Data Unit (EDTPDU)

A TPDU which, on a transport connection, will not be subject to queuing, resource restrictions or any other impediment to its rapid delivery.

4.12 Purge

A mechanism to remove all TSDUs from a transport connection. Purge is analogous to Reset in the Network layer.

4.13 Reference

An identifier selected by a transport entity to uniquely identify the local end of a transport connection. The Reference also identifies locally the transport connection within the transport entity. References are not to be confused with TCEP-IDs.

During connection establishment, References are exchanged between the transport entities involved. Data are sent using the Reference of the receiver.

4.14 Expedited Transport Service Data Units (EDTSDU)

A TSDU which will not be subject to queuing or normal data flow control in its transport connection.

4.15 Network Reset Indication

A signal from the Network layer to the Transport layer indicating that a failure has occured within the Network layer of a type that can be recovered at Transport layer without the need for disconnection of the Network connection.

The properties of the Network layer with respect to Reset indication are the following:

- The Network layer will provide a Reset indication at both ends of the connection in the event of a recoverable Network layer failure.
- The Network layer will ensure that all data sent prior to the Reset indication will be delivered either before the Reset indication or never.
- The Network layer will ensure that data sent after a Reset indication will be delivered at the remote end of the connection after the corresponding remote Reset Indication.

4.16 Network Clear Indication

A signal from the Network layer to the Transport layer indicating disconnection of the network connection.
In addition to providing for Network layer disconnection, this indication may also be used to indicate that a failure has occured within the Network layer that cannot be recovered by the Transport layer except by establishing a new network connection.

4.17 Transport Service Access Point Identifier (TSAP-ID)

The means whereby a Transport Service Access Point having a Transport Address is identified within the Transport layer.

5. TRANSPORT SERVICE

5.1 Objectives of the Transport Service

This Standard is designed in accordance with the ISO Reference Model of Open Systems Interconnection: the overall architecture, the structuring principles and terminology defined therein are to be considered as part of this ECMA Transport Protocol. It is a fundamental principle of this architecture that the data processing oriented functions, i.e. the users of the transport service are clearly decoupled from the communication oriented functions, i.e. providers of transport service (Fig. 1). More precisely, it is the ultimate purpose of the Transport layer to provide two communicating session entities with transport services, i.e. the means for transparent and reliable end-to-end transfer irrespective of the underlying communications media (layer 1 to 3) used. The main requirements for the transport service, to be provided by a transport entity to the local transport user, i.e. the session entity, are:

- Transparency. The transport service shall be transparent, i.e. not restrict the content, format or coding of the user information (data or control), nor ever need to understand its structure or meaning. However each TSDU is structured so that it contains an integral number of octets.

- Error Free. The transport service shall assure error-free delivery. Only exceptional errors are to be visible to the transport service user.

- Network Independence. The transport service shall be homogeneous, while allowing a suitable wide variety of underlying communications media, protocols and mechanisms.

- End-to-End Significance. The transport service shall have end-to-end significance, connecting the end users irrespective of the number of individual communications links used.

- Cost Effectiveness. The transport service shall attempt to optimize the use of the available communication resources to provide the performance required by each communicating transport user at minimum cost.

- Address Decoupling. The transport service shall use a system of addressing particular to it, which is mapped onto the addressing scheme of the supporting communications medium. A transport user, i.e. a session entity, is known to the transport entity only by its transport address. Transport addresses can be used by session entities to uniquely identify other session entities.

This chapter describes the transport services provided by the transport entity to the local session entity across the transport interface. This service is connection oriented. A transaction oriented service is for further study.

The transport entity makes these services available by making use of the underlying services provided by the supporting layers. The definition of the transport service presented here is in terms of service primitives. In many implementations there will exist an equivalent set of interface primitives, but such interface primitives are much affected by the properties of the local operating system and language support and so are not suited to a general description and are not the subject of this Standard. Such a set of interface primitives might include strictly local functions, such as listening for or awaiting an incoming call; these have no exact parallel in the messages passing across the network, nor do they require end-to-end agreement. For this reason the definition of interface primitives as well as the mapping of these primitives onto the transport protocol elements is considered a matter of local implementation in particular systems and it is not the subject of this Standard.

There is not necessarily a one-to-one correspondence between service primitives and TPDUs.
The essential transport service provided to the communicating session entities is the delivery of transport service data units (TSDUs) via transport connections established between pairs of transport addresses identifying the corresponding transport users, i.e. session entities. Users of the transport service are provided with the means to establish, maintain and terminate transport connections which represent a two way simultaneous data path between them.

This transport service is considered to be a basic transport service, the quality of which may be selected on a per-connection basis.

Quality of transport service includes the selection of parameter values and the selection of options.
Parameters, specifying the quality of transport service are e.g.: throughput transit delay, residual error rate, etc.
Options, specifying the quality of transport service are e.g.: purge service available, expedited service available etc.

The transport services provided to the communicating session entities can be considered in three groups, each associated with the three phases of a transport connection: the establishment phase, the data transfer phase and the termination phase.

5.2 Transport Service Primitives

The following transport service primitives are defined:

- Establishment Phase

  Connection Request
  Connection Indication
  Connection Established
  Accept Connection

- Data Transfer Phase

  Data Transfer
  Expedited Data Transfer
  Purge Request/Purge Indication

- Termination Phase

  Disconnect Request
  Disconnect Indication

5.3 Transport Connection Establishment Services

Transport connections can be dynamically established across the transport interface. More than one transport connection can be established between the same pair of transport addresses.

The means by which the session entity can distinguish the transport connection end points are provided at the time of connection establishment by the transport entity in terms of the transport connection end point identifier. This identifier has only local significance.

In the following section transport connection establishment is described by way of an example of the establishment of a transport connection from an originating transport user A to destination transport user B.

5.3.1 Connection Request

Session entity A requests the establishment of a transport connection to session entity B by passing a Connection Request across the transport interface. The Connection Request primitive conveys the following items:

- Source and destination transport addresses.
- Indication of the quality of transport service required (This includes selection of options, e.g. expedited and/or purge services - see 5.4.3 and 5.4.2).
- A limited amount of optional user data (e.g. session control information) which is considered to be a complete TSDU.

5.3.2 Connection Indication

Session entity B receives a Connection Indication as a result of the request by the corresponding session entity A. This assumes that the connection establishment attempt was not aborted by either session entity A or any supporting entity.

The Connection Indication contains the source transport address, the TCEP-ID and the data, if any, generated by session entity A. The quality of service parameters, however, are not necessarily the same as those generated by the originating session entity A.

During the life time of this connection, the TCEP-ID is used for local identification of this connection.

5.3.3 Accept Connection

The destination session entity may either accept the connection request by passing an Accept Connection across the transport interface or refuse the connection request by passing a Disconnect Request (see 5.5). The Accept Connection may also contain a quality of service indication, equal to or less than the received quality of service indication, and optional user data. At this point session entity B is in the data transfer phase.

5.3.4 Connection Established

Session entity A will receive a Connection Established after session entity B issues the Accept Connection.

Session entity A is now in the data transfer phase and connection establishment is complete.

The Connection Established contains the local TCEP-ID, the user data, if any, and an indication of the quality of service negotiated.

5.4 Data Transfer Services

As a basic facility a transport connection consists of a two way simultaneous data path providing for the exchange of transport service data units (TSDUs). Thus a transport connection can be modelled as a pair of independent one way pipes between two users of the transport service. If requested, an additional pair of data pipes is provided on the same transport connection for the exchange of Expedited Transport Service Data Units (EDTSDUs). EDTSDUs are subject to different transport service characteristics and separate flow control.

5.4.1 Data Transfer

The transport service receives transport service data units (TSDUs) at a transport address at one end of a transport connection, and delivers them to the associated transport address at the other end of that same transport connection.

A TSDU has a distinct beginning and end. The transport service maintains the integrity of individual TSDUs. This does not imply that TSDUs exist as such within the transport service. The transport layer does not impose any constraints on the maximum size of TSDUs.

TSDUs will be delivered in the same order in which they were submitted.

A TSDU can be transferred across the transport interface in one or more transport interface data units (TIDUs). The size of a TIDU is not necessarily the same at both ends of a transport connection, i.e. it is dependent on the local implementation.

Data may be held within the transport service until a TSDU delimiter is passed across the transport interface by the sending session entity. It is expected that a TSDU will be received in its entirety by the session entity before it is acted upon. This does not preclude some TIDUs from being handed over to the receiving session entity before a complete TSDU has been received from the sending session entity.

5.4.1.1 Flow Control

The behaviour of the session entity influences and is influenced by transport flow control behaviour by back pressure in the following manner.

A receiving session entity can dynamically control the rate at which it receives TSDUs on a transport connection. This flow control condition may eventually be propagated to the remote transport interface to control the rate at which the sending transport entity will accept TSDUs from the corresponding session entity.

A session entity that is authorized to place a TSDU into the transport pipe cannot necessarily assume that the distant session entity has authorized the acceptance of data.

Similarly, lack of authorization to place a TSDU into the transport pipe does not necessarily imply that the distant session entity has not authorized the acceptance of data.

The session entity may use this mechanism to suspend the transfer of data for an extended period of time.

5.4.1.2 Error Notification

Only data that the receiving transport entity believes to be error free will be passed to the receiving session entity. When errors are detected and all possible recovery attempts in accordance with the agreed quality of service have failed, the connection is terminated and the session entity is notified by a Disconnect Indication. (see 5.5.2).

5.4.2 Purge Request/Purge Indication

This service is available only when the Purge Service has been requested. Invocation of the purge service at one end of a transport connection, i.e. a Purge Request transferred across the transport interface by the session entity, causes the transport service to remove all TSDUs and all EDTSDUs for both directions of a transport connection. The corresponding session entity at the other end of the transport connection will be notified by means of a Purge Indication that a purge has occurred. No TSDU or EDTSDU submitted prior to the invocation of the purge facility is delivered after the notification that the purge has occurred.

The purge service may be used by the transport user to unblock the flow of TSDUs in case of congestion of the transport connection and to resynchronize the use of the transport connection. Invocation of the purge service results in loss of transport user data and consequently requires recovery mechanisms to be effected at transport user level.

The purge service is available during the data transfer phase only. Invocation of purge during any other phase of a transport connection shall be ignored by the transport service. A Purge Indication received after transfer of a Purge Request across the transport interface should be considered by the requesting session entity as a local completion of the purge.

5.4.3 Expedited Data Transfer

This service is available only when Expedited Service has been requested. Expedited Transport Service Data Units (EDTSDUs) provide an additional means for the exchange of up to 8 octets of data between session entities. Since the flow of EDTSDUs is decoupled from the flow of TSDUs it may be used to bypass the flow control mechanism applying to the transfer of TSDUs. As compared to purge (see 5.4.2) which is needed to recover from a blocked flow of normal TSDUs, the expedited data flow may be convenient for the exchange of session status information or for back track mechanisms in the session layer requiring "selective" cancelling of TSDUs.

The transfer of EDTSDUs is subject to a different kind of flow control. An EDTSDU will never be delivered after a normal TSDU that was subsequently submitted by the originating session entity. EDTSDUs are guaranteed to be delivered without loss and in sequence.

5.5 Transport Connection Termination Services

Termination of a transport connection may be initiated at the transport interface either by the session entity issuing a Disconnect Request or by the local transport entity transferring a Disconnect Indication across the transport interface. A transport connection may be terminated in any phase, i.e. establishment phase or data transfer phase. Termination of a transport connection may result in aborting any current data transfer and purge operations at the transport interface.

The termination service does not guarantee the delivery of data that precedes the transport termination and is still in transit. It is the responsibility of the session layer to provide the means for achieving orderly termination of a session before initiating termination of the corresponding transport connection.

5.5.1 Disconnect Request

Either session entity may initiate termination of a transport connection by passing a Disconnect Request across the transport interface. The transport entity will terminate the connection immediately without making sure that TSDUs or EDTSDUs submitted prior to the Disconnect Request have been delivered to the corresponding transport entity. The transport connection will be terminated regardless of the action taken by the distant session entity.

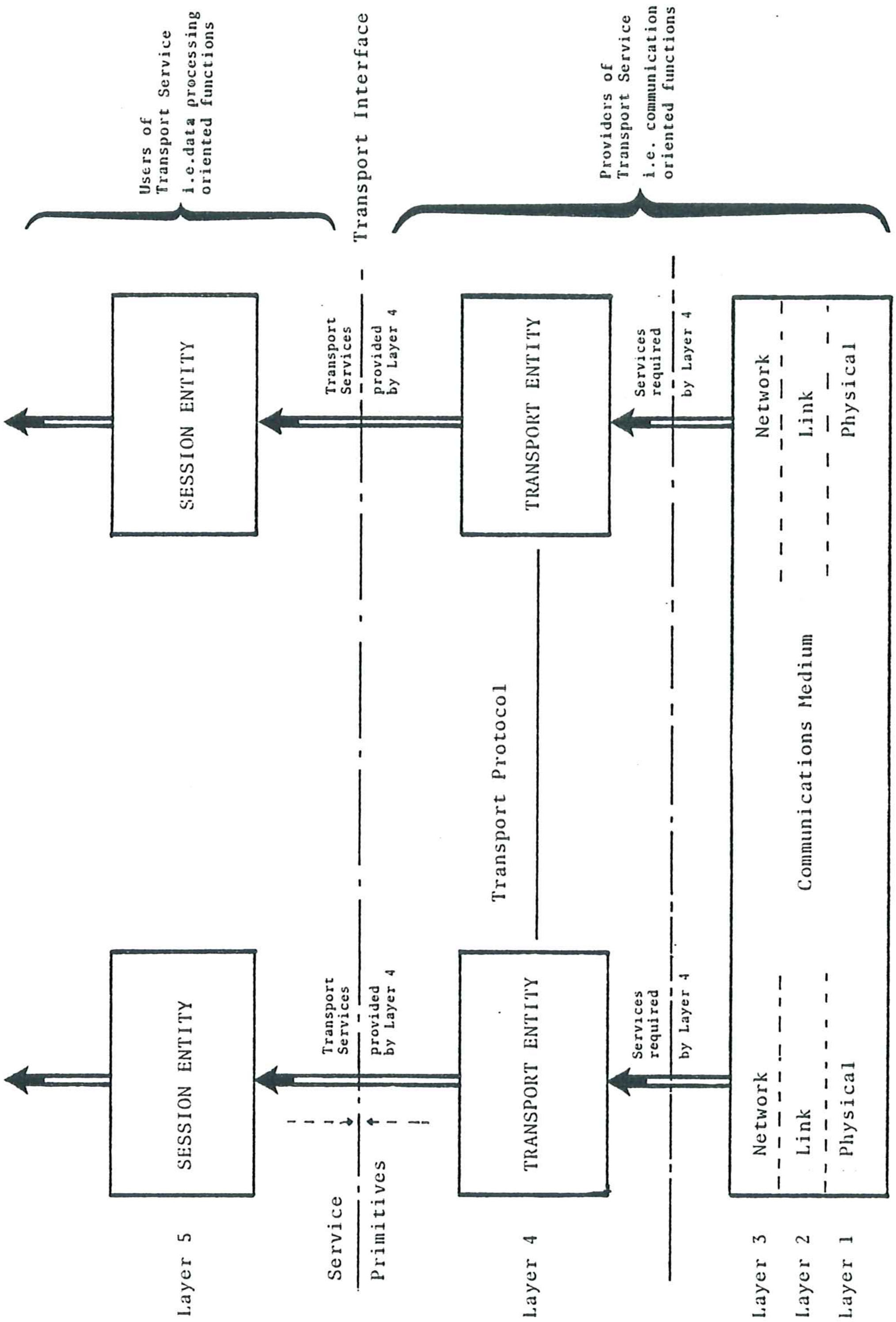
Upon receipt of a Disconnect Request the local transport entity will release the resources and the local connection end-point identifier associated with the transport connection.

NOTE 1

User data with Disconnect Request is for further study.

5.5.2 Disconnect Indication

A Disconnect Indication will be issued by the local transport entity as a result of a Disconnect Request at the remote transport interface or, due to an unrecoverable error detected by the transport system implying that the requested quality of service can no longer be guaranteed, or in response to a Connection Request because of a lack of resources.


Users of
Transport Service
i.e.data processing
oriented functions
Transport Interface
Providers of
Transport Service
i.e. communication
oriented functions
Layer 5
SESSION ENTITY
SESSION ENTITY
Service
Primitives
Transport
Services
provided
by Layer 4
Layer 4
TRANSPORT ENTITY
TRANSPORT ENTITY
Transport Protocol
Services
required
by Layer 4
Layer 3
Layer 2
Layer 1
Network
Link
Physical
Communications Medium

6. FUNCTIONS OF THE TRANSPORT LAYER

6.1 Overview

The functions in the Transport layer are at least those necessary to bridge the gap between the services available from the network layer and those to be offered to the transport users.

The functions in the Transport layer are concerned with the enhancement of quality of service, including all aspects of cost optimization.

6.1.1 Connection Establishment Phase Functions

The goal of the connection establishment phase is to establish a transport connection between two transport users. The functions of Transport layer during this phase are those of selecting References and of matching the requested class of service with the service provided by the Network layer. These are:

- Selection of References: each transport entity is responsible for selecting the Reference which the partner will use. This mechanism is symmetrical and therefore avoids the need to assign a status of master or slave to partners and avoids call collision. This mechanism also provides identification of the transport connection independent of the network connection. The range of References used for transport connections, in a given transport entity, is a local system parameter. Reference ZERO is reserved for the destination reference of the Connect Request, and should therefore not be assigned to a transport connection.
- Selection of the network service that best matches the user requirement, taking into account the charges for the various network services.
- Deciding whether to multiplex multiple transport connections onto a single network connection.
- Establishing optimum TPDU size.
- Selecting the functions that will be operational upon entering the data phase.
- Mapping transport addresses onto network addresses.

6.1.2 Data Transfer Phase

The purpose of the data transfer phase is to transport TSDUs between the two session entities connected by the transport connection. Dependent on the services negotiated during the connection establishment phase, some or all of the following functions may be invoked.

- concatenation: this function is used to collect several TPDUs into a single NSDU. The destination transport entity separates the concatenated TPDUs forming a NSDU.

- segmenting and reassembling: these are the functions of splitting a single data TSDU into a number of TPDUs which are reassembled into their original format at the destination.
- multiplexing: the function used to share a single network connection between two or more transport connections.
- flow control: the function used to regulate the flow of TPDUs between two transport entities on one transport connection.
- error detection: the function used to detect protocol errors and the loss, corruption, duplication or misordering of TPDUs.
- error recovery: the function used to recover from detected and signalled errors.
- expedited data: the function used to bypass the flow control of normal data TPDUs. The flow of expedited data TPDUs is controlled by separate flow control.
- TSDU delimiting: the function used to determine the beginning and ending of a TSDU.
- purge: the function used to flush all data and expedited data in both directions of a given transport connection.

6.1.3 Connection Termination Phase

The function used to provide a non-graceful termination of a transport connection, regardless of the current activity.

6.2 Classes and Options

All functions of the protocol as listed in section 6.1 have been organized in classes and options.

6.2.1 Classes of Functions

A class defines a set of functions. In this protocol five classes are defined:

Class 0 : Simple Terminal Class
Class 1 : Basic Class
Class 2 : Flow Control Class
Class 3 : Error Recovery Class
Class 4 : Error Detection and Recovery Class

Each higher class is a superset of the lower numbered classes.

Note that with the exception of Class 0 transport connections of different class may be multiplexed together to provide different kinds of service using the same network connection. Typical cases would mix the flow controlled classes 2, 3 and 4 bút not normally with class 1.

6.2.2 Options within Classes

Options define additional functions which may be associated with any class.
No option has been defined in this edittion of the Standard. Use of BCC, CRC, etc. to detect data corruption is for further study.

6.2.3 Negotiation

Classes and options within classes are negotiated during the connection establishment phase.

The initiator of the transport connection may indicate during the connection phase the set of options (class of protocol, option, TPDU size etc.) he is willing to support. The acceptor will reply, indicating a set which is equal or less than those indicated, or else he will disconnect in the event that more options are wanted. Once the initiator receives the confirmation with the acceptor's set of options, he may accept it or refuse it by disconnecting.

Default values are defined for certain parameters. One current example is: TPDU size default value = 128 octets.

When such a parameter is not specified either in the request or the confirmation, the default value is assumed. However, the general rule explained above has to be followed.

Parameters which do not have default values have to be indicated both in the request and in the confirmation. Again, for negotiation the general rule has to be followed.

7. CLASS 0 - SIMPLE TERMINAL CLASS

7.1 Objectives of Class 0

The objective of this class is to provide the simplest type of transport connection. It has been defined by CCITT and is being used to support TELETEX terminals connected to switched networks.

7.2 Functionality of Class 0

This class is designed to have minimum functionality.
It provides only the facilities needed for connection establishment with negotiation, data transfer with segmenting and protocol error reporting.
Class 0 contains no facilities for multiplexing, disconnection, flow control, error detection, error recovery, purge or expedited data transfer.

7.3 Functions wihin Class 0

7.3.1 Connection Establishment Phase

This class provides a means to establish a transport connection using a Connection Request command and a Connection Confirm Response. This exchange provides:

- A way to indicate the calling and called session entities. When the network address implicitly defines the transport address (one session entity per network address) these parameters are omitted.
- A way to negotiate selected class and options.
- A way to negotiate optional Data TPDU size.
- A connection identification. The transport connection is identified by use of References.

7.3.2 Data Transfer Phase

The following functions shall be provided:

- Segmenting

  In this class segmenting is possible. It is assumed that the network always preserves the order of the information. Consequently only one indicator called Transport Service Data Unit End Mark is provided.

- Detection and indication of invalid or unimplemented TPDUs.

7.3.3 Connection Termination Phase

There is no explicit transport connection clearing procedure for this class. The life time of the transport connection is directly correlated to the life time of the network connection. A clear of the network connection results in the clearing of the transport connection. The transport user is informed via a Disconnect Indication.

7.4 Connection Establishment Procedures for Class 0

In Class 0 only the transport entity which initiates the network connection may initiate a transport connection.

7.4.1 References

See 6.1.1

7.4.2 Connection Request (CR)

The transport entity shall indicate a transport connection request by transferring a Connection Request (CR) command to the remote transport entity. The reference selected by the transport entity is included in the header of the CR command. The Connection Request (CR) command includes as optional parameters the transport addresses of the calling and called session entities and information for the negotiation on the transport connection being established.

7.4.3 Incoming Connection Request (CR)

On receipt of an incoming Connection Request (CR) command the called transport entity shall assign a Reference to the transport connection resulting in two References, one generated by the source transport entity and one generated by the destination transport entity. The two References are thereafter referred to as "paired References".

Receipt of an incoming Connection Request (CR) command by the calling transport entity shall result in a TPDU ERROR (ERR) command being sent to notify the called transport entity of the procedure error.

7.4.4 Connection Confirm (CC)

The called transport entity shall indicate its acceptance of the transport connection by sending to the calling transport entity a Connection Confirm (CC) response specifying the Reference to be associated with that of the Connection Request (CR) command.

The Connection Confirm (CC) response shall include the transport parameters for negotiation on the transport connection being accepted. CC implicitly acknowledges data, if any, which accompanied CR.

7.4.5 Incoming Connection Confirm (CC)

The receipt of a Connection Confirm (CC) response by the calling transport entity specifying, as destination Reference, the same Reference used as source Reference in the Connection Request (CR) indicates that the transport connection has been established.

7.4.6 Unsuccessful Connections

If a transport connection cannot be established, the called transport entity shall respond to the Transport Connection Request TPDU with a Transport Disconnection Request. The clearing reason shall indicate why the connection was not accepted.

7.5 Data Transfer Procedures

7.5.1 General

The data transfer procedure described in the following subsections applies only when the Transport layer is in the data transfer phase, that is after completion of Transport Connection establishment.

NOTE 2

When a connection is broken, Transport Data TPDUs may be discarded. Hence it is left to the Transport Service User to define protocols able to cope with the various possible situations that may occur.

7.5.2 Transport Data TPDU Length

For Class 0 the standard maximum transport data TPDU length is 128 octets including the data TPDU header octets.

Other optional maximum data field lengths may be supported in conjunction with the transport data TPDU size negotiation function (see 6.2.3). Optional maximum data TPDU lengths shall be chosen from the following list: 256, 512, 1024 and 2048 octets.

The data field of Data TPDUs may contain any number of octets up to the agreed maximum negotiated at connection time.

7.5.3 Transport Service Data Unit (TSDU) End Mark

If a transport entity wishes to indicate a sequence of more than one TPDU, it uses a TSDU End Mark.

In each Data TPDU a transport entity may indicate that no more data will follow; this indication has the effect that such a TPDU will not be combined with subsequent Data TPDU.

Two categories of Data TPDU are defined:

- Category 1, having the TSDU End Mark set to ONE. These TPDUs may or may not have the maximum data field length.
- Category 2, having the TSDU End Mark set to ZERO. These TPDUs have the maximum data field.

Category 1 TPDU shall not be combined with subsequent TPDUs.

A complete TPDU sequence is defined as being composed of either a single category 1 TPDU or consecutive category 2 TPDU followed by a category 1 TPDU. The sequence shall not be preceded by a category 2 TPDU.

7.6 Connection Clearing Procedures for Class 0

There is no explicit Transport Connection clearing procedure for Class 0. A clearing of the network connection shall result in the clearing of the transport connection. The transport user is informed via a Disconnect Indication.

7.7 Treatment of Invalid or Unimplemented TPDUs

At any time, a transport entity may send a TPDU ERROR (ERR) command to report to the remote terminal the receipt of a TPDU which is invalid or not implemented. No confirmation is required to be issued by the terminal following the receipt of a TPDU ERROR (ERR). On receipt of a TPDU ERROR (ERR) the transport entity shall clear the network connection (see 7.6).

8. CLASS 1 - BASIC CLASS

8.1 Objectives of Class 1

The objective is to provide a simple transport connection with minimal overheads suitable for permanent as well as for switched network connections. The class would typically be used for unsophisticated terminals, and when no multiplexing onto network connections is required. However, it can also be used for multiplexing in particular cases when the absence of individual connection flow control is acceptable; e.g. for transport connections with non critical response time requirements, or with infrequent short bursts of traffic and a predictable low combined level of network connection utilisation.

8.2 Functionality of Class 1

Class 1 provides the additional facility to terminate a transport connection independently of the supporting network connection.
Class 1 provides transport connections without flow control, error detection, error recovery, purge or expedited data transfer.

If the network resets or clears the transport connection is terminated without the transport clearing sequence and the transport user is informed.

NOTE 3:

If the user cannot absorb new data and if the appropriate buffers are not available, flow control must be performed at network level. This may involve inefficient use of buffer resources, etc. and delay the traffic on other multiplexed transport connections.

8.3 Functions within Class 1

Class 1 functions include Class 0 functions plus the following:

- User Data in the Connection/Disconnection Phases.

  Class 1 provides the possibility to convey data in the Connection Request and Comfirm commands plus disconnect request commands.

- Connection Identification

  In Class 1 each TPDU conveys a destination Reference. This uniquely identifies the transport connection within the receiving transport entity and thus allows:

  . Multiplexing

  . Independence of lifetime of a transport connection compared to the lifetime of the supporting network connection(s).

  Due to the lack of transport flow control in Class 1, multiplexing in this class should be limited to cases where the user of a transport connection is not in a position to saturate the network connection and associat-

ed buffers for its own purposes.
If overflow occurs, i.e.; if the user cannot absorb new data and if appropriate buffers at transport level are not available flow control must be performed at network level. Therefore overflow on one transport connection will result in slowing down the flow on the other transport connections.

- Concatenation

  A function used to collect several TPDUs into a single NSDU. The destination transport entity separates the concatenated TPDUs forming a NSDU.

- Clearing

  The clearing of a transport connection is achieved in this class either by exchange of Disconnect Request and Disconnect Confirm, or after a Clear (or Reset) of the network connection.

8.4 Connection establishment procedures for Class 1

8.4.1 References

See 6.1.1 for Reference assignment.

Receipt of any TPDU with a Reference that is not assigned to a transport connection other than a Disconnect Request (DR) or Connect Request (CR) shall be ignored. Receipt of a Disconnect Request (DR) for an unassigned Reference shall result in a Disconnect Confirm (DC) response.

8.4.2 Connection Request (CR)

As 7.4.2 but with the addition of optional user data.

NOTE 4:

There may exist a limitation in the length of the optional user data because that length may not match the finally negotiated TPDU size.

8.4.3 Incoming Connection Request (CR)

On receipt of an incoming Connection Request (CR) command the called transport entity shall assign a Reference to the transport connection resulting in two References, one generated by the source transport entity and one generated by the destination transport entity. The two References are thereafter referred to as "paired References".

8.4.4 Connection Confirmed (CC)

As 7.4.4 but with the addition of optional user data.

NOTE 5:

The length of the optional user data is limited by the fact that the CC TPDU containg this data may not exceed the negotiated TPDU size.

8.4.5 Incoming Connection Confirmed

As 7.4.5.

8.4.6 Unsuccessful Connection

As 7.4.6.

8.5 Data Transfer Procedure for Class 1

8.5.1 General

The data transfer procedures described in the following section apply independently to each transport connection existing between two transport entities.
The data phase ends after a transport entity has sent or received a Disconnect Request (DR). The transport entity shall ignore any incoming TPDU except DC or DR.

8.5.2 Transport Data TPDU Length

See 7.5.2.

8.5.3 Transport Service Data Units (TSDU) End Mark

See 7.5.3.

8.5.4 Concatenation

All TPDUs carry in octet 1 a TPDU header length indicator (see 12). Additionally TPDUs are classified as either Data TPDUs or Control TPDUs. Control TPDUs may or may not contain a data field. For TPDUs containing data the length of the data field is indicated by the length of the NSDU.

These provisions permit any number of control TPDUs without data to be concatenated with a single Control TPDU containing data or data TPDUs. The control TPDUs without data, shall precede the TPDU with data, if any.
The number of TPDUs so concatenated is limited by the length of the NSDU.

Concatenation rules are dependent on the freedom that may be exercised when defining the length of the user data field of an NSDU. Where freedom exists then the length of an NSDU should be chosen consistent with the TPDU concatenation rules that it is intended to implement. Where freedom does not exist it is recommended that the length of a Data TPDU be chosen to allow concatenation of at least one AK TPDU with a Data TPDU. However, no constraint is imposed in this Standard on the actual concatenation rules that may be implemented within the constraints of NSDU data field length. Such rules can only properly be defined given knowledge of the parameters relating to particular user systems and their environments.

The concatenated TPDUs may be for different transport connections.

8.6 Connection Clearing Procedures for Class 1

For Class 1 the following procedures are defined enabling transport connections to be cleared independently of the underlying network connection.

8.6.1 Disconnect Request

A transport entity may initiate the clearing of a transport connection by transferring a Disconnect Request (DR) command, with optional user data, indicating the clearing reason to the remote transport entity.

When the remote transport entity is prepared to free the Reference, it shall respond with a Disconnect Confirm (DC) response specifying the paired References. The References are now unassigned.
It is possible that subsequent to transferring a Disconnect Request (DR) command the transport entity will receive other types of TPDU before receiving a Disconnect Confirm (DC) response. Such TPDUs shall be ignored by the transport entity (see 8.6.3).

8.6.2 Incoming Disconnect Request

The remote transport entity shall initiate clearing by sending a Disconnect Request (DR). On receipt of a Disconnect Request (DR) the local transport entity shall respond by transferring a Disconnect Confirm (DC) response. The local Reference is now unassigned.

8.6.3 Disconnect Collision

Disconnect collison occurs when both transport entities simultaneously transfer a Disconnect Request (DR) command. Both transport entities shall consider the clearing completed on receipt of an Incoming Disconnect Request and shall not transfer a Disconnect Confirm (DC) response.

8.7 Action on Detection of a Protocol Error

Detection by a receiving Transport entity of a Protocol. Error indicates the possibility of a serious malfunction of either sending or receiving Transport entities. As such it is an event calling for higher level recovery action.

In addition to the reporting of the error to the local error management function the Transport entity may take any of the following further actions:

- Freeze the state of the Transport Connection for diagnostic purposes.
- Initiate a Disconnect procedure in the expectation that the procedure will not be completed.
- Transmit a TPDU error TPDU in the expectation that it may not be correctly received and reported to the remote higher level error management function.

Action following receipt of a TPDU ERR TPDU is dependant on the receiving system management function.

9. CLASS 2 - FLOW CONTROL CLASS

9.1 Objectives of Class 2

The objective of Class 2 is to provide flow control to help avoid congestion at end-points and on the network connection. Typical use is when traffic is heavy and continuous, or when there is intensive multiplexing. Use of flow control can optimize response times and resource utilisation.

9.2 Functionality of Class 2

A credit mechanism is defined allowing the receiver to inform a sender of the exact amount of data he is willing to receive. Purge and expedited data transfer are available. For treatment of network resets and accidental clearing see 8.2.

9.3 Functions within Class 2

Class 2 functions include Class 1 functions, a flow control mechanism, purge and expedited data transfer.

9.3.1 Numbering of Data TPDU

Each Data TPDU transmitted between transport entities for each direction of transmission in a transport connection shall be sequentially numbered.
Each Data TPDU shall contain a Send Sequence Number T(S).

9.3.2 Flow control principles

The receiver of data TPDUs holds a count of the sequence number of the next expected TPDU. This count is called the Receive Sequence Number, T(R). The receiver indicates to the sender the number of Data TPDUs he is ready to receive by means of a "credit" mechanism. Credits are given using the credit field in the AK TPDU. The value of the credit field, in conjunction with the value of T(R) transported by the YR-TU-NR field of the AK TPDU, is used by the receiver of the AK TPDU to determine whether and how many Data TPDUs may be accepted by the sender of the AK TPDU. Precise definition of flow control principles appears in section 9.5.3.

9.3.3 Use of flow control

Depending on the implementation, credit may reflect the availability of receiving buffers either at transport level or at Transport user level.

Both implementations provide the user with a flow control mechanism i.e. with a means to stop receiving data. In the former flow control is explicit in the latter flow control is performed by back pressure.

9.3.4 Expedited flow

Normal flow is the flow of data subject to the flow control mechanism, expedited flow is the flow of data that the sender may send without explicit agreement of the receiver. This expedited flow has a limited capability and could for example be used to carry session supervisory commands.

The number of expedited data units outstanding at any time is limited to one and the amount of data is limited to 8 octets.

An expedited data may arrive before normal data which was submitted earlier. Normal data submitted after the expedited data will arrive after the expedited data.

9.3.5 Purge

The purge function permits the users of a transport connection to go back to a predefined state, with a possible loss of data, by reinitializing the transport connection.

9.4 Connection Establishment Procedures for Class 2

As for Class 1 (see 8.4), with exchange of initial Credits in Connection Request (CR) and Connection Confirm (CC) as described in section 9.5.

9.5 Data Transfer Procedures for Class 2

On each transport connection the transmission of Data TPDUs is controlled separately for each direction and is based on authorization from the receiver.

This authorization is provided through the use of the TPDU Credit field. Credit field values are only meaningful in the following TPDUs: CR, CC, AK, PR and PC.

If the network resets or clears the network connection, all transport connections are terminated without the transport clearing sequence. The References become unassigned.

9.5.1 Numbering of Data TPDUs

Each Data TPDU transmitted between transport entities for each direction of transmission in a transport connection is sequentially numbered.

The sequence numbering of the TPDU is performed modulo 128. The TPDU sequence numbers cycle through the entire range 0 to 127.

Only Data TPDUs contain a sequence number, called the Send Sequence number T(S).

The first Data TPDU to be transmitted between transport entities for a given direction of data transmission after the connection establishment, shall have Send Sequence number T(S) equal to zero.

Receipt of a Data TPDU whose sequence number is not equal to the expected value T(R) shall be regarded as a protocol error.

9.5.2 Window definition

For each transport connection and for each direction of data transmission a "transmit window" is defined as the ordered set of consecutive data TPDUs authorized to be transmitted in that direction. The lowest sequence number within the transmit window is referred to as the "lower window edge".

The lowest sequence number above the transmit window is referred to as the "upper window edge".

The receiver indicates to the sender via the credit field, CDT, in CR, CC, AK, PR, PC TPDUs that the transmit window of the sender is N data TPDUs starting from and including the data TPDU numbered T(R).

The value of T(R) is explicitly present in the YR-TU-NR field of the AK TPDUs. For CR, CC, PR, PC it has the implicit value of zero.

The receiver must have received the in-sequence TPDU with T(S) = T(R) - 1 (mod. 128) before using T(R) in an AK TPDU to send credit. The value of T(R) at connection establishment and after the purge procedure is zero and the receiver may use this value to send credit.

9.5.3 Flow Control

A number, modulo 128, referred to as a TPDU Receive Sequence Number T(R), conveys between transport entities information that the receiver acknowledges receipt of all Data TPDUs up to but not including the Data TPDU with T(S) equal to T(R). Additionally, when transmitted between transport entities, T(R) becomes the lower window edge. The lower window edge plus the value of the credit field contained on the received TPDU becomes the upper window edge.

When the Send Sequence Number T(S) of the next Data TPDU to be transmitted by a transport entity is within its transmit window, the remote transport entity shall accept this Data TPDU. When the Send Sequence Number T(S) of the next Data TPDU to be transmitted by the transport entity is outside its transmit window, the remote transport entity shall discard this Data TPDU and recognize a protocol error.

When the Send Sequence Number T(S) of the next Data TPDU to be transmitted by the remote transport entity is within its transmit window, the remote transport entity is authorized to transmit this Data TPDU. When the Send Sequence Number T(S) of the next Data TPDU to be transmitted by the remote transport entity is outside its transmit window, this Data TPDU shall not be transmitted.

The TPDU Receive Sequence Number, T(R), is conveyed in the Acknowledge (AK) TPDU.

9.5.4 Reducing the credit

The value of the upper window edge cannot be decreased in this class during the transfer state. If, at a certain point of time, the upper window edge value is U, the reception of an AK TPDU having YR-TU-NR = M and CDT = W such that:

$$M + W \text{ (mod. 128) } < U$$

is a protocol error.

Provided the previous statements are respected, CDT field may take any value including zero.

9.5.5 Procedure for Expedited Data Transfer

The procedure for Expedited Data Transfer allows a transport entity to transmit Data to the remote transport entity without following the flow control procedure of the normal data flow. This procedure can only apply in the transfer phase.

The expedited procedure has no effect on the transfer and flow control applying to normal Data TPDUs.

To transmit Expedited Data, the transport entity sends an Expedited Data TPDU (EDTPDU). The size of the data field is limited to 8 octets. The data field contains a complete EDTSDU. The remote transport will then confirm the receipt of the EDTPDU by transferring an expedited TPDU confirmation (EATPDU). A transport entity can send another EDTPDU only after having received an EATPDU for the previously transmitted EDTPDU. In Class 2 the NR field of the EDTPDU is always zero.

9.5.6 Procedure for Purge

Purge is used to re-initialize a given transport connection. It removes all Data and (if any) EDTPDUs in both directions of the transport connection. The purge procedure can only apply in the Data Transfer Phase. The purge procedure applies independently to different transport connections.

9.5.6.1 Purge Request

After having discarded all TPDUs for the transport connection (any of DT, AK, ED, EA) which have not yet been passed to the network or to the session entity, the transport entity shall indicate a Purge Request for this connnection by transmitting a Purge (PR) TPDU with the appropriate destination Reference to the remote transport entity. On sending the PR command the transport entity shall wait a Purge Confirmation (PC) TPDU for this transport connection. In this state incoming DT, AK, ED or EA TPDUs shall be ignored. The sending of DT or ED TPDUs is not permitted.

9.5.6.2 Incoming Purge Request

On receipt of an incoming Purge Command (PR) for a transport connection, the transport entity shall discard all DT, AK, EA and ED TPDUs for that transport connection which have not yet been passed to the session entity or to the network.

9.5.6.3 Purge Confirmation

After having discarded the TPDUs as above the transport entity shall send a Purge Confirmation (PC) TPDU to the remote transport entity.

9.5.6.4 Incoming Purge Confirmation

The receipt of a Purge Confirmation (PC) TPDU after the sending of a Purge Request indicates to the transport entity that the purge procedure has been completed. The sending of TPDUs appropriate to the Data Transfer Phase (DT, ED) is now permitted.

9.5.6.5 Purge Collision

As both transport entities may simultaneously request purge, purge collision can occur. Therefore, an incoming Purge command (PR) occuring while the transport entity is awaiting Purge Confirmation for a particular transport connection shall be considered as Purge Confirmation.

9.5.6.6. Effect on Flow Control

On sending a Purge (PR) or a Purge Confirmation (PC) command the transport entity shall reset the send and receive counters of the corresponding transport connection to zero. Thus, the first DT command sent after the purge shall carry the TPDU number zero and the next DT command expected shall contain TPDU number zero.

All previously granted credits are cancelled by the Purge procedure. The window sizes for the forthcoming data transfer are determined by the credit fields exchanged with the PR and PC commands. TPDUs discarded during the Purge procedure do not have any impact on send or receive counters, discarded Data TPDUs are not acknowledged by AK.

9.6 Connection Clearing Procedures for Class 2

The clearing procedure for Class 2 is identical to the clearing procedure for Class 1. See section 8.6 for details.

9.7 Treatment of Invalid or Unimplemented TPDUs

See 8.7.

10. CLASS 3 - ERROR RECOVERY CLASS

10.1 Objectives of Class 3

The objective of Class 3 in addition to those of Class 2 is to mask errors indicated by the network. Selection of this Class is usually based upon reliability criteria. The assumptions made concerning the network layer behaviour with respect to error indication appear in 4.15 and 4.16.

10.2 Functionality of Class 3

This Class provides the functionality of Class 2 plus the ability to recover after a failure signalled by the Network layer without involving the user of the transport service. The mechanisms used to achieve this functionality also allow the implementation of more flexible flow control.

10.3 Functions within Class 3

Class 3 functions include Class 2 functions and the ability to recover after a failure signalled by the network without informing the user of the transport connection.

10.3.1 Error Recovery Principles

The sending transport entity keeps a copy of transmitted Data TPDUs and ED TPDUs until it receives a positive acknowledgement which allows copies to be released. It may also receive a RJ command inviting it to retransmit, or transmit, all Data TPDUs, if any, from the point in the sequence indicated in the RJ command.

This is especially the case, when a failure is indicated by the network (Reset, for example). The transport entity sends a RJ command in order to indicate the sequence number of the next expected TPDU.
Error recovery for ED TPDUs is achieved by retransmission (see 10.5.4).

10.3.2 Relationship between Flow Control and Error Recovery

Acknowledgement is performed by use of the T(R) count. A credit is associated with this acknowledgement which may be equal to or greater than zero. Thus it is possible to acknowledge data without giving the right to send new data.

10.4 Connection Establishment Procedures for Class 3

The rules for Class 2 shall apply with the addition of the following rules which apply on receipt of an error indication from the Network layer.

- Reception of an error indication by a transport entity, which, prior to this event, has sent a CR and has not yet received a CC, shall cause the transport entity to retransmit CR.

- Reception of an error indication by a transport entity which, prior to this event, has sent a CC and has not since received any TPDUs shall cause the transport entity to wait for reception of CR, RJ or DR TPDU. In this case:

- Reception of CR shall cause the transport entity to retransmit CC.
- Reception of RJ shall cause the transport entity to transmit an RJ with a YR-TU-NR equal to zero and enter the data phase.
- Reception of a DR shall cause termination of the transport connection as for Classes 1 and 2 (see 8.4).

10.5 Data Transfer Procedures for Class 3

The receipt of an AK TPDU with YR-TU-NR equal to T(R) confirms that the remote transport entity has correctly received all Data TPDU(s) up to but not including TPDU with number T(R).

NOTE 6:

This allows the receiver of the AK to release the buffers containing the acknowledged Data TPDUs.

10.5.1 Reject condition with T(R)

TPDU retransmission is a facility which allows a transport entity to request retransmission of one or several consecutive Data TPDUs from the remote transport entity. A transport reject condition is signalled to the remote transport entity by transmission of an RJ TPDU whose YR-TU-NR field indicates the sequence number of the next expected Data TPDU.

On detection of this condition the remote transport entity shall consider the TPDUs following and including the TPDU numbered T(R) as not yet sent. These TPDUs, if any, shall be retransmitted following the normal flow control rules. Transport entity may send credit, while rejecting, by means of the CDT field, the RJ TPDU.
Receipt of a RJ acknowledges all previously sent TPDU(s) up to and not including T(R).

The transport entity may not specify a T(R) in the RJ TPDU less than that which has previously been acknowledged. Receipt of an RJ TPDU with a T(R) which has been previously acknowledged shall be considered a protocol error.

Additional DT TPDUs pending initial transmission may follow the retransmitted DT TPDU(s) if the window is not closed.

10.5.2 Reducing the credit

It is possible to decrease the value of the upper window edge down to the sequence number transported by the YR-TU-NR field of the RJ TPDU.

NOTE 7:

In such a case the credit equal to zero achieves the effect of a Receive not Ready condition.

10.5.3 Behaviour after an error signalled by the Network layer

After receiving an error indication from the Network layer, the transport entity shall transmit to the remote transport entity an RJ TPDU with YR-TU-NR field indicating the sequence number of the next expected Data TPDU.

After transmitting this RJ TPDU, the transport entity shall await receipt of the RJ TPDU similarly transmitted by the remote transport entity before transmitting data. Some Data TPDUs may then be retransmitted as required according to section 10.5.1.

10.5.4 Procedure for Expedited Data Transfer

In Class 3, the ED TPDU NR field of the Expedited Data (ED) TPDU contains an identification number. This number must be different for successive ED TPDUs. That is, when an ED TPDU has been sent and an EA TPDU for the ED TPDU has been received, the next ED TPDU must have a different value in the NR field of the ED TPDU.

No other significance is attached to this field. It is recommended, but not essential, that the values used be consecutive modulo 128. When a transport entity receives an ED TPDU for a transport connection, it shall respond by transmitting an Expedited Acknowledgement (EA) TPDU. It places in the YR-TU-NR field the value contained in the NR field of the received ED TPDU. If and only if this value is different from the NR field of the previously received ED TPDU, the data contained in the TPDU is to be passed to the session entity.

If an error indication from the Network layer is received before the receipt of the expected Expedited Acknowledgement (EA) TPDU, the transport entity shall retransmit the ED TPDU with the same value in the ED TPDU N(R) field. By the rule described in the previous paragraph, the session entity does not receive data corresponding to the same expedited TPDU more than once.

10.5.5 Procedure for Purge

The rules for Class 2 apply with the following additions:

- RJ TPDUs received during the purge procedure shall be ignored.
- On receipt of an incoming Purge command (PR) RJ TPDUs which have not yet been passed to the network shall be discarded.
- If an error indication from the Network layer is received during the purge procedure the initiator of the procedure shall re-initiate it.

10.6 Connection Clearing Procedures for Class 3

The rules for Classes 1 and 2 apply with the addition of the following rule:

Receipt of an error indication by a transport entity, which, prior to this event has sent a DR, shall cause this transport entity to retransmit DR. Only DC shall be accepted and interpreted as the completion of the connection clearing sequence. The related Reference shall become unassigned. In the event of clearing collision, delivery of optional user data included with the Disconnect Request (DR) to the session entity is not guaranteed.

10.7 Treatment of Invalid or Unimplemented TPDUs

At any time, a transport entity may send a TPDU ERROR (ERR) command to report to the remote terminal the receipt of a TPDU which is invalid or not implemented. No confirmation is required to be issued by the terminal following the receipt of a TPDU ERROR (ERR).

11. CLASS 4 ERROR DETECTION AND RECOVERY CLASS

As stated in the Introduction (see 1.2) this issue of the Standard is published without a definition of Class 4 procedures. This deficiency will be rectified in a later edition. It is the intention of ECMA that, when fully defined, Class 4 procedures will be fully compatible with the other Classes and will be a superset of these procedures in conformance with the general philosophy that has been adopted for this Standard.

The additional objectives of Class 4 over and above those of Class 3 will be to meet the requirements of datagram and similar networks in which sequence integrity is not maintained. The requirements of resilient systems requiring high integrity of data transfer will be specifically addressed.

12. FORMAT OF COMMANDS AND RESPONSES

12.1 Summary

This section defines the detailed encoding of the commands and responses defined in the previous sections.
The table below summarizes the commands and responses and defines their allocation to the classes.

NOTE 8:

Class 4 is not defined in this edition of the Standard.

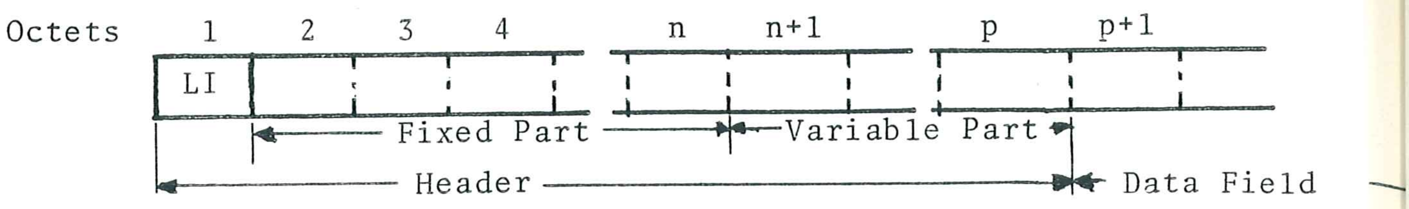
	Classes					Section	Code
	0	1	2	3	4		
CR Connection Request	x	x	x	x		12.3	1110
CC Connection Confirm	x	x	x	x		12.4	1101
DR Disconnect Request	x	x	x	x		12.5	1000
DC Disconnect Confirm		x	x	x		12.6	1100
DT Data	x	x	x	x		12.7	1111
ED Expedited Data			x	x		12.8	0001
AK Data Acknowledgement			x	x		12.9	0110
EA Expedited Data Acknowledgement			x	x		12.10	0010
RJ Reject				x		12.11	0101
PR Purge Request			x	x		12.12	1010
PC Purge Confirm			x	x		12.13	1001
ERR TPDU Error	x	x	x	x		12.14	0111

12.2 Structure

As defined in the previous sections, all the Transport Protocol Data Units (TPDU) shall contain an integral number of octets. The octets in a TPDU are numbered consecutively starting from 1, and are transmitted in ascending order. The bits in an octet are numbered from 1 to 8, where bit 1 is the low-order bit and it is transmitted first.

There are two types of TPDUs:

- Data TPDUs, used to transfer Transport Service Data Units (TSDUs). The structure of the TSDUs is maintained by means of the TSDU End Mark.
- Control TPDUs, used to control the transport protocol functions, including the optional functions.



A TPDU is divided into four parts:

- Length Indicator Field (LI)
- Fixed Part
- Variable Part
- Data Field

The Length Indicator Field, Fixed part and Variable part constitute the Header of the TPDU.

12.2.1 <u>Length Indicator Field</u>

This field is contained in the first octet of the TPDUs. The length is indicated by a binary number, with a maximum value of 254 (11111110). The length indicated is the header length, including parameters, but excluding the length indicator field and user data, if any.
The value 255 (11111111) is reserved for possible extensions.

12.2.2 <u>Fixed Part</u>

The fixed part contains frequently occuring functions including the code of the TPDU. The length and the structure of the fixed part are defined by the TPDU code, defined by bits 5 to 8 of the second octet of the header.

12.2.3 <u>Variable Part</u>

The variable part is used to define parameters relating to optional functions. It may contain one or more parameters. Each parameter is coded as follows:

8 7 6 5 4 3 2 1	Bits
Parameter Code	Octet n+1
Parameter Length Indicator	n+2
Parameter Value	n+3

- The parameter code field is coded in binary and, without extensions, provides a maximum number of 255 parameters.
  Parameter code 11111111 is reserved for possible extensions of the parameter code.

- The parameter length indicator indicates the length, in octets, of the parameter value field. The length is indicated by a binary number with a maximum value of 255.

- The parameter value field contains the value of the parameter identified in the parameter code field.

12.2.4 Data Field

This field contains transparent user data. Restrictions on its size are defined for each command.

12.3 Connection Request (CR)

12.3.1 Structure

1	2	3	4	5	6	7	8	p	p+1
LI	CR 1110 ¦ CDT	00000000	00000000	SOURCE-REFERENCE		Class Options	PARAMETERS		User Data

12.3.2 LI

See 12.2.1.

12.3.3 Fixed Part (Octets 2 to 7)

- CR: Connection Request Code 1110
- CDT: Initial Credit Allocation (Set to 0000 in Classes 0 and 1)
- SOURCE-REFERENCE: Reference selected by the transport entity initiating the CR TPDU to identify the requested transport connection
- CLASSES: Bits 8-5 of octet 7 defines the transport protocol Class to be operated over the requested transport connection. This field may take one of the following values.

0000 : Class 0
0001 : Class 1
0010 : Class 2
0011 : Class 3
0100 : Class 4

Options: Bits 4-1 of octet 7 define the standard optional services to be made available as part of the requested transport connection. This field is set to 0000 in this edition of the Standard.

12.3.4 Variable Part (Octets 8 to p)

The following parameters are permitted in the variable part:

- Transport Service Access Point Identifier (TSAP-ID)

Parameter code: 11000001 for the identifier of the Calling TSAP
11000010 for the identifier of the Called TSAP

NOTE 9:

To fully meet the objectives of OSI, Transport Addresses should be made subject to Global addressing conventions. The Global authority needed to establish and administer such conventions has yet to be established in ISO. The structure and encoding of Address parameter value fields has therefore been left undefined.

However, for the Teletex application of Class 0 procedures CCITT have defined that Transport Addresses will be encoded using International Alphabet No. 5.

The term Transport Service Access Point Identifier has been chosen as a parameter name to avoid any pre-empting of any final decisions that may ultimately be taken with respect to Global Transport Addressing. The TSAP-ID parameter is the means whereby a TSAP is identified within the Transport Protocol. It is not to be confused with the TSAP Address as this is known to both Transport and Session entities.

If a TSAP-ID is given in the request it has to be returned in the confirmation.

- TPDU size

  This parameter defines the proposed maximum TPDU size (in octets including the header) to be used over the requested transport connection. The coding of this parameter is:

  Parameter Code 11000000

  Parameter Value field

  Class 0

00001011	2048 octets
00001010	1024 octets
00001001	512 octets
00001000	256 octets
00000111	128 octets

  Other classes: a binary number indicating the number of octets.

12.3.5 User Data (Octets p+1 to end)

No user data are permitted in Class 0, and are optional in the other classes.

12.4 Connection Confirm (CC)

12.4.1 Structure

1	2		3	4	5	6	7	8		p	p+1
LI	CC 1101	CDT	DST-REFERENCE		SOURCE-REFERENCE		Class Options	PARAMETERS			User Data

12.4.2 LI

See 12.2.1.

12.4.3 Fixed Part (Octets 2 to 7)

CC: Connection Confirm Code = 1101

CDT: Initial Credit Allocation (Set to 0000 in Classes 0 and 1).

DST-REFERENCE: Reference identifying the requested Transport connection at the remote transport entity.

SOURCE-REFERENCE: Reference selected by the Transport entity initiating the CC TPDU to identify the confirmed transport connection.

CLASSES: Defines the selected transport protocol class to be operated over the accepted transport connection according to the negotiation rules specified in 6.2.3.

12.4.4 Variable Part (Octets 8 to p)

See 12.3.4.

12.4.5 User Data (Octets p+1 to end)

See 12.3.5.

12.5 Disconnect Request (DR)

12.5.1 Structure

1	2	3	4	5	6	7	8 ... p	p+1
LI	DR 10000000	DST-REFERENCE		SOURCE-REFERENCE		Reason	PARAMETERS	User Data

12.5.2 LI

See 12.2.1.

12.5.3 Fixed Part (Octets 2 to 7)

DR: Disconnect Request Code : 1000

DST-REFERENCE: Reference identifying the transport connection at the remote transport entity.

SOURCE-REFERENCE: Reference identifying the transport connection at the transport entity initiating the command. Value zero when no references are assigned.

REASON: Defines the reason for disconnecting transport. This field shall take one of the following values:

128 + 0 - Normal disconnect initiated by session control

128 + 1 - Remote transport entity congestion at connect request time

128 + 2 - Connection negotiation failed

128 + 3 - Duplicate connection detected

128 + 4 - Mismatched references

128 + 5 - Protocol Error

128 + 6 - Destination Session Entity specified not available

128 + 7 - Reference overflow

128 + 8 - Connection Request refused on this network connection

255 - Unknown Reason

For Teletex, the following values have been defined by CCITT:

0 : Reason not specified

1 : Terminal occupied

2 : Terminal out of order

3 : Address unknown

12.5.4 Variable Part (Octets 8 to p)

A parameter may be provided to allow additional information related to the clearing of the connection.

Parameter Code : 11100000

Parameter Value field : additional information.

12.5.5 User Data (Octets p+1 to end)

Contains a limited amount of Data for complementary information. If the disconnection is the result of difficulties in data transfer, transport of these Data is not guaranteed.

If no agreed TPDU size exists and the receiver of DR has proposed no TPDU size, DR user data field should be less or equal to 16 octets. No user data are permitted in Class 0.

12.6 Disconnect Confirm (DC) (Not used in Class 0)

12.6.1 Structure

1	2	3	4	5	6
LI	DC 1100 ⁞ 0000	DST-REFERENCE		SOURCE-REFERENCE	

12.6.2 LI

See 12.2.1.

12.6.3 Fixed Part (Octets 2 to 6)

DC: Disconnect Confirm Code : 1100

DST-REFERENCE: See 12.3.3.

SOURCE-REFERENCE: See 12.4.3.

12.7 Data (DT)

12.7.1 Structure

Reduced Format for Class 0

1	2	3	4	5
LI	DT 1111 0000	EOT TPDU NR	User Data	...

Normal Format for Classes 1 to 4

1	2	3	4	5	6
LI	DT 1111 0000	DST-REFERENCE		EOT TPDU NR	User Data ...

12.7.2 LI

See 12.2.1.

12.7.3 Fixed Part (Class 0: - Octets 2 to 3, Class 1-4: - Octets 2 to 5)

DT: Data Transfer Code: 1111

DST-REFERENCE: See Section 12.4 (not applicable in Class 0)

EOT: When set to ONE, indicates that the current DT TPDU contains the last Data Unit of a Data Unit sequence (End of TDSU).

TPDU-NR: TPDU Send Sequence Number. This field must be set to zero in the Classes 0 and 1.

12.7.4 Variable Part

Not applicable.

12.7.5 User Data Field (Class 0: - Octets 4 to end, Class 1-4: Octets 6 to end)

This field contains data of the TSDU being transmitted. The length of this field is limited to the negotiated TPDU size for this transport connection minus 3 octets in Class 0, and minus 5 octets (for the header) in the other classes.

12.8 Expedited Data (ED) (not used in Classes 0 and 1)

12.8.1 Structure

1	2	3	4	5	6
LI	ED 0001 0000	DST-REFERENCE		1 EDTPDU NR	User Field ...

12.8.2 LI

See 12.2.1.

12.8.3 Fixed Part (Octets 2 to 5)

ED: Expedited Data command code 0001

DST-REFERENCE: Same as 12.4

ED TPDU NR: Expedited TPDU identification number (not applicable in Class 2)

12.8.4 Variable Part

Not applicable.

12.8.5 User Data Field (Octets 6 to end)

Up to 8 octets.

12.9 Data Acknowledgement (AK) (not applicable for Classes 0 and 1)

12.9.1 Structure

1	2		3	4	5
LI	AK 0110	CDT	DST-REFERENCE		0 YR-TU-NR

12.9.2 LI

See 12.2.1.

12.9.3 Fixed Part (Octets 2 to 5)

AK: Acknowledgement Command Code 0110

CDT: Credit value

DST-REFERENCE: See 12.5.3

YR-TU-NR: Sequence number indicating the next expected DT TPDU number (New lower window edge) (Not applicable in Class 2)

12.10 Expedited Data Acknowledgement (EA) (Not applicable for Classes 0 and 1)

12.10.1 Structure

1	2	3	4	5
LI	EA 00100000	DST-REFERENCE		0 YR-TU-NR

12.10.2 LI

See 12.2.1.

12.10.3 Fixed Part (Octets 2 to 5)

EA: Acknowledgement Command Code 0010

DST-REFERENCE: See 12.5.3

YR-TU-NR: identification of the EDTPDU being acknowledged

12.11 Reject (RJ) (Not applicable for Classes 0, 1 and 2)

12.11.1 Structure

1	2		3	4	5
LI	RJ 0101	CDT	DST-REFERENCE		0 YR-TU-NR

12.11.2 LI

See 12.2.1.

12.11.3 Fixed Part (Octets 2 to 5)

RJ: Reject Command Code 0101

CDT: Credit value

DST-REFERENCE: See 12.4.3

YR-TU-NR: Sequence number indicating the next expected TPDU from which retransmission should occur.

12.12 Purge Request (PR) (Not applicable for Classes 0 and 1)

12.12.1 Structure

1	2		3	4	5
LI	PR 1010	CDT	DST-REFERENCE		Reason

12.12.2 LI

See 12.2.1

12.12.3 Fixed Part (Octets 2 to 5)

PR: Purge Request Code 1010

CDT: New Credit Allocation

DST-REFERENCE: See 12.4.3

Reason: 0000001 = purge requested by user.

12.13 Purge Confirmation (PC) (Not applicable for Classes 0 and 1)

12.13.1 Structure

1	2		3	4	5
LI	PC 1001	CDT	DST-REFERENCE		Reserved

12.13.2 LI

See 12.2.1.

12.13.3 Fixed Part (Octets 2 to 5)

PC: Purge Confirm Code 1001

CDT: New Credit allocated

DST-REFERENCE: See 12.4.3.

12.14 TPDU Error (ERR)

12.14.1 Structure

1	2	3	4	5	6
LI	ERR 0111 0000	DST-REFERENCE		Reject Cause	PARAMETERS

12.14.2 LI

See 12.2.1

12.14.3 Fixed Part (Octets 2 to 5)

ERR: TPDU Error Code 0111

DST-REFERENCE: See 12.4.3

REJECT CAUSE:

00000000	Reason not specified
00000001	Invalid parameter code
00000010	Invalid TPDU type
00000011	Invalid parameter value

12.14.4 Variable Part (Octets 6 to end)

Parameter Code 11000001

Parameter Value Field: contains the bit pattern of the rejected TPDU up to and including the octet which caused the rejection. This parameter is mandatory in Class 0.