Network Working Group
Request for Comments: 994

ANSI X3S3.3 86-80
ISO TC97/SC6/N 3998
March 1986

I S O
INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/TC 97/SC 6
TELECOMMUNICATIONS AND INFORMATION
EXCHANGE BETWEEN SYSTEMS
Secretariat: USA (ANSI)

Title: Final Text of DIS 8473, Protocol for Providing the Connectionless-mode Network Service

Source: DIS 8473 Editor

Contents

List of Tables

List of Figures

0 Introduction

This Protocol Standard is one of a set of International Standards produced to facilitate the interconnection of open systems. The set of standards covers the services and protocols required to achieve such interconnection.

This Protocol Standard is positioned with respect to other related standards by the layers defined in the Reference Model for Open Systems Interconnection (ISO 7498). In particular, it is a protocol of the Network Layer. This Protocol may be used between network-entities in end systems or in Network Layer relay systems (or both). It provides the Connectionless-mode Network Service as defined in Addendum 1 to the Network Service Definition Covering Connectionless-mode Transmission (ISO 8348/AD1).

The interrelationship of these standards is illustrated in Figure 1 below:

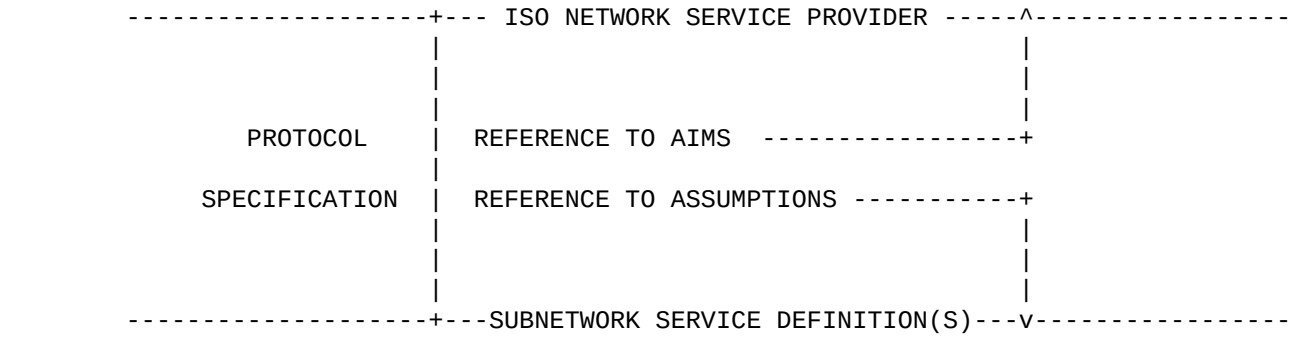

Figure 1: Interrelationship of Standards

1 Scope and Field of Application

This International Standard specifies a protocol which is used to provide the Connectionless-mode Network Service as described in Addendum 1 to the Network Service Definition Covering Connectionless-mode Transmission. The protocol relies upon the provision of an underlying connectionless-mode service by real subnetworks and/or data links. The underlying connectionless-mode service assumed by the protocol may be obtained either directly, from a connectionless-mode real subnetwork, or indirectly, through the operation of an appropriate Subnetwork Dependent Convergence Function (SNDCF) or Protocol (SNDCP) over a connection-mode real subnetwork as described in ISO 8648, Internal Organization of the Network Layer.

This Standard specifies:

a) procedures for the connectionless transmission of data and control information from one network-entity to a peer network-entity;

b) the encoding of the protocol data units (PDUs) used for the transmission of data and control information, comprising a variable-length protocol header format;

c) procedures for the correct interpretation of protocol control information; and

d) the functional requirements for implementations claiming conformance to the Standard.

The procedures are defined in terms of:

a) the interactions among peer network-entities through the exchange of protocol data units;

b) the interactions between a network-entity and a Network Service user through the exchange of Network Service primitives; and

c) the interactions between a network-entity and an underlying service provider through the exchange of service primitives.

2 References

ISO 7498, Information Processing Systems --- Open Systems Interconnection --- Basic Reference Model

DIS 7498/AD1, Information Processing Systems --- Open Systems Interconnection --- Addendum to ISO 7498 Covering Connectionless-mode Transmission

ISO 8348, Information Processing Systems --- Telecommunications and Information Exchange between Systems --- Network Service Definition

ISO 8348/AD1, Information Processing Systems --- Telecommunications and Information Exchange between Systems --- Addendum to the Network Service Definition Covering Connectionless-mode Transmission

ISO 8348/AD2, Information Processing Systems --- Telecommunications and Information Exchange between Systems --- Addendum to the Network Service Definition Covering Network Layer Addressing*

DIS 8648, Information Processing Systems --- Telecommunications and Information Exchange between Systems --- Internal Organization of the Network Layer

ISO 8509, Technical Report --- OSI Service Conventions

ISO 9074, A Formal Description Technique based on an Extended State Transition Model

---

*At present, at the stage of Draft; publication anticipated in due course.

SECTION ONE. GENERAL

3 Definitions

3.1 Reference Model Definitions

This document makes use of the following concepts defined in ISO 7498:

(a) End system

(b) Network entity

(c) Network layer

(d) Network protocol

(e) Network protocol data unit

(f) Network relay

(g) Network service

(h) Network service access point

(i) Network service access point address

(j) Routing

(k) Service

(l) Service data unit

3.2 Service Conventions Definitions

This Protocol Standard makes use of the following terms from the OSI Service Conventions Technical Report (ISO TR 8509):

(a) Service provider

(b) Service user

3.3 Network Layer Architecture Definitions

This Protocol Standard makes use of the following terms from the Internal Organization of the Network Layer (ISO 8648):

(a) Intermediate system

(b) Relay system

(c) Subnetwork

3.4 Network Layer Addressing Definitions

This Protocol Standard makes use of the following terms from ISO 8348/AD2, Addendum to the Network Service Definition Covering Network Layer addressing:

(a) Network addressing domain

(b) Network protocol address information

(c) Subnetwork point of attachment

3.5 Additional Definitions

For the purposes of this Protocol Standard, the following definitions apply:

(a) derived PDU --- a protocol data unit whose fields are identical to those of an initial PDU, except that it carries only a segment of the user data from an N-UNITDATA request.

(b) initial PDU --- a protocol data unit carrying the whole of the userq data from an N-UNITDATA request.

(c) local matter --- a decision made by a system concerning its behavior in the Network Layer that is not prescribed or constrained by this Protocol Standard.

(d) network-entity title --- an identifier for a network-entity which has the same abstract syntax as an NSAP address, and which can be used to unambiguously identify a network-entity in an end or intermediate system.

(e) reassembly --- the act of regenerating an initial PDU from two or more derived PDUs.

(f) segment --- a distinct unit of data consisting of part or all of the user data provided in the N-UNITDATA request and delivered in the N-UNITDATA indication.

(g) segmentation --- the act of generating two or more derived PDUs from an initial or derived PDU. The derived PDUs together carry the entire user data of the initial or derived PDU from which they were generated.

Note:
It is possible that such an initial PDU will never actually be generated for a particular N-UNITDATA request, owing to the immediate application of segmentation.

4 Symbols and Abbreviations

4.1 Data Units

NSDU	Network Service Data Unit
PDU	Protocol Data Unit
SNSDU	Subnetwork Service Data Unit

4.2 Protocol Data Units

DT PDU	Data Protocol Data Unit
ER PDU	Error Report Protocol Data Unit

4.3 Protocol Data Unit Fields

CS	Checksum
DA	Destination Address
DAL	Destination Address Length
DUID	Data Unit Identifier
E/R	Error Report Flag
LI	Length Indicator
LT	Lifetime
MS	More Segments Flag
NLPID	Network Layer Protocol Identifier
SA	Source Address
SAL	Source Address Length
SL	Segment Length
SO	Segment Offset
SP	Segmentation Permitted Flag
TL	Total Lengt
TP	Type
V/P	Version/Protocol Identifier Extension

4.4 Parameters

DA	Destination Address
QOS	Quality of Service
SA	Source Address

4.5 Miscellaneous

CLNP	Connectionless-mode Network Protocol
NS	Network Service
NPAI	Network Protocol Address Information
NSAP	Network Service Access Point
SDU	Service Data Uni
SN	Subnetwork
SNDCF	Subnetwork Dependent Convergence Function
SNDCP	Subnetwork Dependent Convergence Protocol
SNICP	Subnetwork Independent Convergence Protocol
SNPA	Subnetwork Point of Attachment

5 Overview of the Protocol

5.1 Internal Organization of the Network Layer

The architectural organization of the Network Layer is described in a separate document, Internal Organization of the Network Layer (ISO 8648). ISO 8648 identifies and categorizes the way in which functions can be performed within the Network Layer by Network Layer protocols, thus providing a uniform framework for describing how protocols operating either individually or cooperatively in the Network Layer can be used to provide the OSI Network Service. This protocol is designed to be used in the context of the internetworking protocol approach to the provision of the Connectionless-mode Network Service defined in that Standard.

This protocol is intended for use in the Subnetwork Independent Convergence Protocol (SNICP) role. A protocol which fulfills the SNICP role operates to construct the OSI Network Service over a defined set of underlying services, performing functions which are necessary to support the uniform appearance of the OSI Connectionless-mode Network Service over a homogeneous or heterogeneous set of interconnected subnetworks. This protocol is defined to accommodate variability where Subnetwork Dependent Convergence Protocols and/or Subnetwork Access Protocols do not provide all of the functions necessary to support the Connectionless-mode Network Service over all or part of the path from one NSAP to another.

As described in ISO 8648, a protocol at the Network Layer may fulfill different roles in different configurations. Although this protocol is designed particularly to be suitable for a SNICP role in the context of the internetworking protocol approach to the provision of the Connectionless-mode Network Service, it may also be used to fulfill other roles and may therefore be used in the context of other approaches to subnetwork interconnection.

The specification of this protocol begins with a definition of the underlying service which it assumes. This service is made available by the operation of other Network Layer protocols or through provision of the Data Link Service. The underlying service assumed by this protocol is described in Clause 5.5.

5.2 Subsets of the Protocol

Two proper subsets of the full protocol are defined which permit the use of known subnetwork characteristics and are therefore not subnetwork independent.

The Inactive Network Layer protocol subset is a null-function subset which can be used when it is known that the source and destination end-systems are connected by a single subnetwork, and when none of the functions performed by the full protocol is required to provide

the Connectionless-mode Network Service between any pair of end-systems.

The Non-segmenting protocol subset permits simplification of the header where it is known that the source and destination end-systems are connected by subnetworks whose service data unit sizes are greater than or equal to a known bound which is large enough so that segmentation is not required. This subset is selected by setting the Segmentation Permitted flag to zero.

5.3 Addresses and Titles

The following Clauses describe the addresses and titles used by this Protocol.

5.3.1 Addresses

The Source Address and Destination Address parameters referred to in Clause 7.3 of this International Standard are OSI Network Service Access Point Addresses. The syntax and semantics of an OSI Network Service Access Point Address are described in a separate document, ISO 8348/AD2, Addendum to the Network Service Definition Covering Network Layer Addressing.

The encoding used by this protocol to convey NSAP Addresses shall be the preferred binary encoding specified in ISO 8348/AD2; the entire NSAP address, taken as a whole, is represented explicitly as a string of binary octets. This string is conveyed in its entirety in the address fields described in Clause 7.3. The rules governing the generation of the preferred binary encoding are described in ISO 8348/AD2.

5.3.2 Network-entity Titles

A network-entity title is an identifier for a network-entity in an endsystem or intermediate-system. Network-entity titles are allocated from the same name space as NSAP addresses, and the determination of whether an address is an NSAP address or a network-entity title depends on the context in which the address is interpreted. The entries in the Source Routing and Recording of Route parameters defined in Clauses 7.5.4 and 7.5.5 are network-entity titles. The Source Address and Destination Address parameters in the Error Report PDU defined in Clause 7.9.1.2 are also network-entity titles.

The encoding used by this protocol to convey network-entity titles shall also be the preferred binary encoding; again, the entire network-entity title, taken as a whole, is represented explicitly as a string of binary octets. This string is conveyed in its entirety in the fields described in Clauses 7.5.4, 7.5.5, and 7.9.1.2.

5.4 Service Provided by the Network Layer

The service provided by this protocol is the Connectionless-mode Network Service described in ISO 8348/AD1, Addendum to the Network Service Definition Covering Connectionless-mode Transmission. The Network Service primitives provided are summarized in Table 1:

PRIMITIVES		PARAMETERS
N_UNITDATA	.Request .Indication	N_Source_Address, N_Destination_Address, N_Quality_of_Service, N_Userdata

Table 1: Service Primitives for Underlying Service

The Addendum to the Network Service Definition Covering Connectionless-mode Transmission (ISO 8348/AD1) states that the maximum size of a connectionless-mode Network-service-data-unit (NSDU) is limited to 64512 octets.

5.5 Underlying Service Assumed by the Protocol

The underlying service required to support this protocol is defined by the following primitives:

PRIMITIVES		PARAMETERS
SN_UNITDATA	.Request .Indication	SN_Source_Address, SN_Destination_Address, SN_Quality_of_Service, SN_Userdata

Table 2: Service Primitives for Underlying Service

Note:

These service primitives are used to describe the abstract interface which exists between the ISO 8473 protocol machine and an underlying real subnetwork or a Subnetwork Dependent Convergence Function which operates over a real subnetwork or real data link to provide the required underlying service.

5.5.1 Subnetwork Points of Attachment

The source and destination addresses specify the points of attachment to a public or private subnetwork(s) involved in the transmission. Subnetwork Point of Attachment addresses (SNPAs) are defined by each individual subnetwork authority.

The syntax and semantics of SNPAs are not defined in this Standard.

5.5.2 Subnetwork Quality of Service

Subnetwork Quality of Service describes aspects of an underlying connectionless-mode service which are attributable solely to the underlying service.

Associated with each connectionless-mode transmission, certain measures of Quality of Service are requested when the primitive action is initiated. These requested measures (or parameter values and options) are based on a priori knowledge of the service(s) made available to it by the subnetwork. Knowledge of the nature and type of service available is typically obtained prior to an invocation of the underlying connectionless-mode service.

The Quality of Service parameters identified for the underlying connectionless-mode service may in some circumstances be directly derivable from or mappable onto those identified in the Connectionless-mode Network Service. The following parameters as defined in ISO 8348/AD1, Addendum to the Network Service Definition Covering Connectionlessmode Transmission, may be employed:

(a) transit delay;

(b) protection against unauthorized access;

(c) cost determinants;

(d) priority; and

(e) residual error probability.

Note:
For those subnetworks which do not inherently provide Quality of Service as a parameter when the primitive action is initiated, it is a local matter as to how the semantics of the service requested might be preserved. In particular, there may be instances in which the Quality of Service requested cannot be maintained. In such circumstances, an attempt shall be made to deliver the protocol data unit at whatever Quality of Service is available.

5.5.3 Subnetwork User Data

The SN-Userdata is an ordered multiple of octets, and is transferred transparently between the specified subnetwork points of attachment.

The underlying service assumed by the CLNP is required to support a service data unit size of at least 512 octets.

If the minimum service data unit sizes supported by all of the subnetworks involved in the transmission of a particular PDU are known to be large enough that segmentation is not required, then the Non-segmenting protocol subset may be used.

5.5.4 Subnetwork Dependent Convergence Functions

Subnetwork Dependent Convergence Functions may be performed to provide an underlying connectionless-mode service in the case where a real subnetwork does not inherently provide the connectionless-mode service assumed by the protocol. If a subnetwork inherently provides a connection-mode service, a Subnetwork Dependent Convergence Function provides a mapping into the required underlying service. Subnetwork Dependent Convergence Functions may also be required in those cases where functions assumed from the underlying service are not performed. In some cases, this may require the operation of an explicit protocol (i.e., a protocol involving explicit exchanges of protocol control information between peer network-entities) in the Subnetwork Dependent Convergence Protocol (SNDCP) role. However, there may also be cases where the functionality required to fulfill the SNDCP role consists simply of a set of rules for manipulating the underlying service.

5.6 Service Assumed from Local Environment

A timer service must be provided to allow the protocol entity to schedule events.

There are three primitives associated with the S-TIMER service:

1. the S--TIMER Request,
2. the S--TIMER Response, and
3. the S--TIMER Cancel.

The S--TIMER Request primitive indicates to the local environment that it should initiate a timer of the specified name and subscript and maintain it for the duration specified by the time parameter.

The S--TIMER Response primitive is initiated by the local environment to indicate that the delay requested by the corresponding S-TIMER Request primitive has elapsed.

The S--TIMER Cancel primitive is an indication to the local environment that the specified timer(s) should be canceled. If the subscript parameter is not specified, then all timers with the specified name are canceled; otherwise, the timer of the given name and subscript is cancelled. If no timers correspond to the parameters specified, the local environment takes no action.

The parameters of the S--TIMER service primitives are specified in Table 3.

PRIMITIVES		PARAMETERS
S--TIMER	.Request	S-Time, S-Name, S-Subscript
	.Response	S-Name, S-Subscript

Table 3: Timer Primitives

The time parameter indicates the time duration of the specified timer. An identifiying label is associated with a timer by means of the name parameter. The subscript parameter specifies a value to distinguish timers with the same name. The name and subscript taken together constitute a unique reference to the timer.

Timers used in association with a specific protocol funtion are defined under that protocol function.

Note:

This International Standard does not define specific values for the timers. Any derivations described in this Standard are not mandatory. Timer values should be chosen so that the requested Quality of Service can be provided, given the known characteristics of the underlying service.

SECTION TWO. SPECIFICATION OF THE PROTOCOL

6 Protocol Functions

This Clause describes the functions performed as part of the Protocol.

Not all of the functions must be performed by every implementation. Clause 6.17 specifies which functions may be omitted, and the correct behavior when requested functions are not implemented.

6.1 PDU Composition Function

This function is responsible for the construction of a protocol data unit according to the rules governing the encoding of PDUs given in Clause 7. Protocol Control Information required for delivering the data unit to its destination is determined from current state and local information and from the parameters associated with the N-UNITDATA Request.

Network Protocol Address Information (NPAI) for the Source Address and Destination Address fields of the PDU header is derived from the NS-Source-Address and NS-Destination-Address parameters. The NS-Destination-Address and NS-Quality-of-Service parameters, together with current state and local information, are used to determine which optional functions are to be selected. User data passed from the Network Service User (NS-Userdata) forms the Data field of the protocol data unit.

During the composition of the protocol data unit, a Data Unit Identifier is assigned to distinguish this request to transmit NS-Userdata to a particular destination NS User from other such requests. The originator of the PDU must choose the Data Unit Identifier so that it remains unique (for this Source and Destination address pair) for the maximum lifetime of the Initial PDU in the network; this rule applies for any PDUs derived from the Initial PDU as a result of the application of the Segmentation Function (see Clause 6.7). Derived PDUs are considered to correspond to the same Initial PDU, and hence the same N-UNITDATA Request, if they have the same Source Address, Destination Address, and Data Unit Identifier.

The Data Unit Identifier is also available for ancillary functions such as error reporting (see Clause 6.10).

The total length of the PDU in octets is determined by the originator and placed in the Total Length field of the PDU header. This field is not changed in any Derived PDU for the lifetime of the protocol data unit.

When the Non-segmenting protocol subset is employed, neither the Total Length field nor the Data Unit Identifier field is present. The rules governing the PDU composition function are modified in this case as follows. During the composition of the protocol data unit, the total length of the PDU in octets is determined by the originator and placed in the Segment Length field of the PDU header. This field is not changed for the lifetime of the PDU. No Data Unit Identification is provided.

6.2 PDU Decomposition Function

This function is responsible for removing the Protocol Control Information from the protocol data unit. During this process, information pertinent to the generation of the N-UNITDATA Indication is determined as follows. The NS-Source-Address and NS-Destination-Address parameters of the N-UNITDATA Indication are recovered from the NPAI in the Source and Destination Address fields of the PDU header. The data field of the PDU received is reserved until all segments of the original service data unit have been received; collectively, these form the NS-Userdata parameter of the N-UNITDATA Indication. Information relating to the Quality of Service provided during the transmission of the PDU is determined from the Quality of Service and other information contained in the Options Part of the PDU header. This information constitutes the NS-Quality-of-Service parameter of the N-UNITDATA Indication.

6.3 Header Format Analysis Function

This function determines whether the full protocol described in this Standard is employed, or one of the defined proper subsets thereof. If the protocol data unit has a Network Layer Protocol Identifier indicating that this is a standard version of the Protocol, this function determines whether a received PDU has reached its destination, using the Destination Address provided in the PDU. If the Destination Address provided in the PDU identifies an NSAP served by this network-entity, then the PDU has reached its destination; if not, it must be forwarded.

If the protocol data unit has a Network Layer Protocol Identifier indicating that the Inactive Network Layer Protocol subset is in use, then no further analysis of the PDU header is required. The network-entity in this case determines that either the Subnetwork Point of Attachment address encoded as network protocol address information in the supporting subnetwork protocol corresponds directly to an NSAP address serviced by this network-entity or that an error has occurred. If the subnetwork protocol data unit has been delivered correctly, then the PDU may be decomposed according to the procedures described for that particular subnetwork protocol.

6.4 PDU Lifetime Control Function

This function is used to enforce the maximum PDU lifetime. It is closely associated with the Header Format Analysis function. This function determines whether a PDU received may be forwarded or whether its assigned lifetime has expired, in which case it must be discarded.

The operation of the PDU Lifetime Control function depends upon the Lifetime field in the PDU header. This field contains, at any time, the remaining lifetime of the PDU (represented in units of 500 milliseconds). The Lifetime of the Initial PDU is determined by the originating network-entity, and placed in the Lifetime field of the PDU. When the Segmentation function is applied to a PDU, the value of the Lifetime field of the Initial PDU is copied into all of the Derived PDUs.

The Lifetime of the PDU is decremented by every network-entity which processes the PDU. When a network-entity processes a PDU, it decrements the PDU Lifetime by at least one. The value of the PDU Lifetime field shall be decremented by more than one if the sum of:

1. the transit delay in the underlying service from which the PDU was received; and

2. the delay within the system processing the PDU

exceeds or is estimated to exceed 500 milliseconds. In this case, the lifetime field should be decremented by one for each additional 500 milliseconds of delay. The determination of delay need not be precise, but where a precise value cannot be ascertained, the value used shall be an overestimate, not an underestimate.

If the Lifetime field reaches a value of zero before the PDU is delivered to the destination, the PDU must be discarded. The Error Reporting function shall be invoked as described in Clause 6.10, Error Reporting Function, and may result in the generation of an Error Report PDU. It is a local matter whether the destination network-entity performs the Lifetime Control function.

6.5 Route PDU Function

This function determines the network-entity to which a protocol data unit should be forwarded and the underlying service that must be used to reach that network-entity, using the Destination Address and the total length of the PDU. Where segmentation is required, the Route PDU function further determines over which underlying service Derived PDUs/segments must be sent in order to reach that network-entity. The results of the Route PDU function are passed to the Forward PDU function (along with the PDU itself) for further processing. Selection of the underlying service that must be used to reach the "next" sys-

tem in the route is initially influenced by the NS-Quality-of- Service parameter of the N-UNITDATA Request, which specifies the QoS requested by the sending NS User. Whether this QoS is to be provided directly by the CLNP, through the selection of the Quality of Service Maintenance parameter and other optional parameters, or through the QoS facilities offered by each of the underlying services is determined prior to invocation of the Forward PDU function. Route selection by intermediate systems may subsequently be influenced by the values of the Quality of Service Maintenance parameter (if present), and other optional parameters (if present).

6.6 Forward PDU Function

This function issues an SN-UNITDATA Request primitive (see Clause 5.5), supplying the subnetwork or SNDCF identified by the Route PDU function with the protocol data unit as user data to be transmitted, the address information required by that subnetwork or SNDCF to identify the "next" system within the subnetwork-specific addressing domain (this may be an intermediate-system or the destination end-system), and Quality of Service constraints (if any) to be considered in the processing of the user data.

When the PDU to be forwarded is longer than the maximum service data user size provided by the underlying service, the Segmentation function is applied (See Clause 6.7, which follows).

6.7 Segmentation Function

Segmentation is performed when the size of the protocol data unit is greater than the maximum service data unit size supported by the underlying service to be used to transmit the PDU.

Segmentation consists of composing two or more new PDUs (Derived PDUs) from the PDU received. The PDU received may be the Initial PDU, or it may be a Derived PDU. All of the header information from the PDU to be segmented, with the exception of the segment length and checksum fields of the fixed part, and the segment offset of the segmentation part, is duplicated in each Derived PDU, including all of the address part, the data unit identifier and total length of the segmentation part, and the options part (if present).

Note:

The rules for forwarding and segmentation guarantee that the header length is the same for all segments (Derived PDUs) of the Initial PDU, and is the same as the header length of the Initial PDU. The size of a PDU header will not change due to operation of any protocol function.

The user data encapsulated within the PDU received are divided such that the Derived PDUs satisfy the size requirements of the user data parameter field of the primitive used to access the underlying ser-

vice.

Derived PDUs are identified as being from the same Initial PDU by means of

(a) the source address,

(b) the destination address, and

(c) the data unit identifier.

Segmentation shall not result in the generation of a Derived PDU containing less than eight (8) octets of user data.

The following fields of the PDU header are used in conjunction with the Segmentation function:

(a) Segment Offset --- identifies, with respect to the start of the Initial PDU, the octet at which the segment begins;

(b) Segment Length --- specifies the number of octets in the Derived PDU, including both header and data;

(c) More Segments Flag --- is set to one if this Derived PDU does not contain, as its final octet of user data, the final octet of the Initial PDU; and

(d) Total Length --- specifies the entire length of the Initial PDU, including both header and data.

Derived PDUs may be further segmented without constraining the routing of the individual Derived PDUs. The Segmentation Permitted flag is set to one to indicate that segmentation is permitted. If the Initial PDU is not to be segmented at any point during its lifetime in the network, the flag is set to zero by the source network-entity. The setting of the Segmentation Permitted flag cannot be changed by any other network-entity for the lifetime of the Initial PDU and any Derived PDUs.

6.8 Reassembly Function

The Reassembly function reconstructs the Initial PDU from the Derived PDUs generated by the operation of the Segmentation Function on the Initial PDU (and, recursively, on subsequent Derived PDUs). A bound on the time during which segments (Derived PDUs) of an Initial PDU will be held at a reassembly point before being discarded is provided, so that reassembly resources may be released when it is no longer expected that any outstanding segments of the Initial PDU will arrive at the reassembly point. Upon reception of a Derived PDU, a reassembly timer is initiated with a value which indicates the amount of

time which must elapse before any outstanding segments of the Initial PDU shall be assumed to be lost. When this timer expires, all segments (Derived PDUs) of the Initial PDU held at the reassembly point are discarded, the resources allocated for those segments are freed, and if selected, an Error Report is generated (see Clause 6.10). While the exact relationship between reassembly lifetime and PDU lifetime is a local matter, the Reassembly Function must preserve the intent of the PDU lifetime. Consequently, the reassembly function must discard PDUs whose lifetime would otherwise have expired had they not been under the control of the reassembly function.

Note:

1. Methods of bounding reassembly lifetime are discussed in Annex B.

2. The Segmentation and Reassembly functions are intended to be used in such a way that the fewest possible segments are generated at each segmentation point and reassembly takes place at the final destination of a PDU. However, other schemes which

   (a) interact with the routing algorithm to favor paths on which fewer segments are generated;

   (b) generate more segments than absolutely required in order to avoid additional segmentation at some subsequent point; or

   (c) allow partial or full reassembly at some intermediate point along the route

   are not precluded. The information necessary to enable the use of one of these alternative strategies may be made available through the operation of a Network Layer Management function or by other means.

3. The originator of the Initial PDU determines the value of the Segmentation Permitted flag in the Initial PDU and all Derived PDUs (if any). Partial or full reassembly in an intermediate system (Note 2 (c) above) cannot change this value in the Initial PDU or any PDU derived from it, and cannot therefore add or remove the segmentation part of the header.

6.9 Discard PDU Function

This function performs all of the actions necessary to free the resources reserved by the network-entity when any of the following situations is encountered (Note: the list is not exhaustive):

(a) A violation of protocol procedure has occurred.

(b) A PDU is received whose checksum is inconsistent with its contents.

(c) A PDU is received, but due to local congestion, it cannot be processed.

(d) A PDU is received whose header cannot be analyzed.

(e) A PDU is received which cannot be segmented and cannot be forwarded because its length exceeds the maximum service data unit size supported by any underlying service available for transmission of the PDU to the next network-entity on the chosen route.

(f) A PDU is received whose destination address is unreachable or unknown.

(g) Incorrect or invalid source routing was specified. This may include a syntax error in the source routing field, an unknown or unreachable address in the source routing field, or a path which is not acceptable for other reasons.

(h) A PDU is received whose PDU lifetime has expired or whose lifetime expires during reassembly.

(i) A PDU is received which contains an unsupported option.

6.10 Error Reporting Function

6.10.1 Overview

This function causes an attempt to return an Error Report PDU to the source network-entity when a protocol data unit is discarded in accordance with Clause 6.9.

The Error Report PDU identifies the discarded PDU, specifies the type of error detected, and identifies the location in the header of the discarded PDU at which the error was detected. At least the entire header of the Discarded PDU (and, at the discretion of the originator of the Error Report PDU none, all, or part of the data field) is placed in the data field of the Error Report PDU.

The originator of a Data PDU may control the generation of Error Report PDUs. An Error Report flag in the original PDU is set by the source network-entity to indicate that an Error Report PDU is to be returned if the Initial PDU or any PDUs derived from it are discarded; if the flag is not set, Error Reports are to be suppressed.

Note:

1. The suppression of Error Report PDUs is controlled by the

originating network-entity and not by the NS User. Care should be exercised by the originator with regard to suppressing ER PDUs so that error reporting is not suppressed for every PDU generated.

2. Non-receipt of an Error Report PDU does not imply correct delivery of a PDU issued by a source network-entity.

6.10.2 Requirements

An Error Report PDU shall not be generated to report the discard of an Error Report PDU.

An Error Report PDU shall not be generated to report the discard of a Data PDU unless that PDU has the Error Report flag set to allow Error Reports.

If a Data PDU is discarded, and the Error Report flag has been set to allow Error Reports, an Error Report PDU shall be generated if the reason for discard is one of the reasons for discard enumerated in Clause 6.9, subject to the conditions described in Clause 6.10.4.

Note:
If a Data PDU with the E/R flag set to allow Error Reports is discarded for any other reason, an ER PDU may be generated (as an implementation option).

6.10.3 Processing of Error Reports

An Error Report PDU is composed from information contained in the header of the discarded Data PDU to which the Error Report refers. The contents of the Source Address field of the discarded Data PDU are used as the Destination Address of the Error Report PDU. This value, which in the context of the Data PDU was used as an NSAP Address, is used in the context of the Error Report PDU as the network-entity title of the network-entity that originated the Data PDU. The network- entity title of the originator of the Error Report PDU is conveyed in the Source Address field of the header of the Error Report PDU. The value of the Lifetime field is determined in accordance with Clause 6.4. Optional parameters are selected in accordance with Clause 6.10.4.

Segmentation of Error Report PDUs is not permitted; hence, no Segmentation Part is present. The total length of the ER PDU in octets is placed in the Segment Length field of the ER PDU header. This field is not changed during the lifetime of the ER PDU. If the originator of the ER PDU determines that the size of the ER PDU exceeds the maximum service data unit size of the underlying service, the ER PDU shall be truncated to the maximum service data unit size (see Clause 5.5.3) and forwarded with no other change. Error Report PDUs are routed and forwarded by intermediate-system network-entities in the

same way as Data PDUs.

Note:
The requirement that the underlying service assumed by the CLNP must be capable of supporting a service data unit size of at least 512 octets guarantees that the entire header of the discarded Data PDU can be conveyed in the data field of any ER PDU.

When an ER PDU is decomposed upon reaching its destination, information that may be used to interpret and act upon the Error Report is obtained as follows. The network-entity title recovered from the NPAI in the Source Address field of the ER PDU header is used to identify the network-entity which generated the Error Report. The reason for generating the Error Report is extracted from the Options Part of the PDU header. The entire header of the discarded Data PDU (and part or all of the original user data) is extracted from the data field of the ER PDU to assist in determining the nature of the error.

6.10.4 Relationship of Data PDU Options to Error Reports

The generation of an Error Report is affected by options that are present in the corresponding Data PDU. The presence of options in the original Data PDU that are not supported by the system which has discarded that PDU may cause the suppression of an Error Report even if the original Data PDU indicated that an Error Report should be generated in the event of a discard.

The processing of an Error Report is also affected by options which are present in the corresponding Data PDU. In particular, options selected for the original Data PDU affect which options are included in the corresponding Error Report PDU. The selection of options for an Error Report PDU is governed by the following requirements:

(a) If the Priority Option or the QoS Maintenance Option is selected in the original Data PDU, and the system generating the Error Report PDU supports the option, then the Error Report PDU shall specify the option.

(b) If the Security Option is selected in the Data PDU, and the system generating the Error Report supports this option, then the Error Report PDU shall specify the option using the value that was specified in the original Data PDU. If the system does not support the Security Option, an Error Report must not be generated for a Data PDU that selects the Security Option.

(c) If the Complete Source Route Option is selected in the original Data PDU, and the system generating the Error Report PDU supports this option, then the error Report shall specify the Complete Source Route option. The Source Route parameter value is obtained by extracting from the original Data PDU that portion of the complete source route that has already been traversed, and reversing the

order of network-entity titles which comprise the list. If the system does not support the Complete Source Route Option, an Error Report must not be generated for a Data PDU that selects the Complete Source Route option.

(d) The Padding, Partial Source Routing, and Record Route Options, if supported, may be specified in the Error Report PDU.

Note:
The values of the optional parameters in (d) above may be derived as a local matter, or they may be based upon the corresponding values in the original Data PDU.

6.11 PDU Header Error Detection

The PDU Header Error Detection function protects against failure of intermediate or end-system network-entities due to the processing of erroneous information in the PDU header. The function is realized by a checksum computed on the entire PDU header. The checksum is verified at each point at which the PDU header is processed. If the checksum calculation fails, the PDU must be discarded. If PDU header fields are modified (for example, due to operation of the lifetime function), then the checksum is modified so that the checksum remains valid.

The use of the Header Error Detection function is optional, and is selected by the originating network-entity. If the function is not used, the checksum field of the PDU header is set to zero.

If the function is selected by the originating network-entity, the value of the checksum field causes the following formulae to be satisfied:

(The Sum from i=1 to L of a(i)) (mod 255) = 0

(The Sum from i=1 to L of (L - i + 1) * a(i)) (mod 255) = 0

where L = the number of octets in the PDU header, and a(i) = the value of the octet at position i. The first octet in the PDU header is considered to occupy position i = 0.

When the function is in use, neither octet of the checksum field may be set to zero.

Note:

1. To ensure that inadvertent modification of a header while a PDU is being processed by an intermediate system (for example, due to a memory fault) may still be detected by the PDU Header Error function, an intermediate system network-

entity must not recompute the checksum for the entire header, even if fields are modified.

2. Annex C contains descriptions of algorithms which may be used to calculate the correct value of the checksum field when the PDU is created, and to update the value of the checksum field when the header is modified.

6.12 Padding Function

The padding function is provided to allow space to be reserved in the PDU header which is not used to support any other function. Octet alignment must be maintained.

Note:
An example of the use of this function is to cause the data field of a PDU to begin on a convenient boundary for the originating network-entity, such as a computer word boundary.

6.13 Security

The provision of protection services (e.g., data origin authentication, data confidentiality, and data integrity of a single connectionless-mode NSDU) is performed by the Security Function.

The Security Function is related to the Protection from Unauthorized Access Quality of Service parameter described in ISO 8348/AD1, Addendum to the Network Service Definition Covering Connectionless-mode Transmission. The function is realized through selection of the security parameter in the options part of the PDU header.

This Standard does not specify the way in which protection services are to be provided; it only provides for the encoding of security information in the PDU header. To facilitate interoperation between end-systems and network relay-systems by avoiding different interpretations of the same encoding, a means to distinguish user-defined security encodings from standardized security encodings is described in Clause 7.5.3.

Note:
As an implementation consideration, data origin authentication may be provided through the use of a cryptographically generated or enciphered checksum (unique from the PDU Header Error Detection mechanism); data confidentiality and data integrity may be provided via route control mechanisms.

6.14 Source Routing Function

The Source Routing function allows the originator to specify the path a generated PDU must take. Source routing may only be selected by the

originator of a PDU. Source Routing is accomplished using a list of network-entity titles held in a parameter within the options part of the PDU header. The length of this parameter is determined by the originating network-entity, and does not change as the PDU traverses the network.

The Source Route parameter includes information used by the originating end-system when determining the initial route of the PDU. Only the titles of intermediate system network-entities are included in the list; the network-entity title of the destination of the PDU is not included in the list.

Associated with the list of network-entity titles is an indicator which identifies the next entry in the list to be used; this indicator is advanced by the receiver of the PDU when the next title in the list matches its own. The indicator is updated as the PDU is forwarded so as to identify the appropriate entry at each stage of relaying.

Two forms of the Source Routing function are provided. The first form, referred to as Complete Source Routing, requires that the specified path must be taken; that is, only those systems identified in the list may be visited by the PDU while en route to the destination, and each system must be visited in the order specified. If the specified path cannot be taken, the PDU must be discarded. Clause 6.10 describes the circumstances in which an attempt shall be made to inform the originator of the discard using the Error Reporting function.

The second form is referred to as Partial Source Routing. Again, each system identified in the list must be visited in the order specified while en route to the destination. However, with this form of source routing the PDU may take any path necessary to arrive at the next intermediate system in the list, which may include visiting intermediate systems that are not identified in the list. The PDU will not be discarded (for source routing related reasons) unless one of the systems specified cannot be reached by any available route.

6.15 Record Route Function

The Record Route function records the path(s) taken by a PDU as it traverses a series of intermediate systems. A recorded route consists of a list of network-entity titles held in a parameter within the options part of the PDU header. The length of this parameter is determined by the originating network-entity, and does not change as the PDU traverses the network.

The list is constructed as the PDU is forwarded along a path towards its destination. Only the titles of intermediate system network-entities are included in the recorded route. The network-entity title of the originator of the PDU is not recorded in the list.

When an intermediate system network-entity processes a PDU containing the Record Route parameter, the system adds its own networkentity title at the end of the list of recorded network-entity titles. An indicator is maintained to identify the next available octet to be used for recording of route. This indicator is updated as entries are added to the list as follows. The length of the entry to be added to the list is added to the value of the next available octet indicator, and this sum is compared with the length of the Record Route parameter. If the addition of the entry to the list would exceed the size of the parameter, the next available octet indicator is set to indicate that route recording has been terminated. The network-entity title is not added to the list. The PDU may still be forwarded to its final destination, without further addition of network-entity titles.

If the addition of the entry would not exceed the size of the Record Route parameter, the next available octet indicator is updated with the new value, and the network-entity title is added to the head of the list after the other entries have been moved.

Two forms of the Record Route function are provided. The first form is referred to as Complete Route Recording. It requires that the list of network-entity titles be a complete and accurate record of all intermediate systems visited by a PDU (including Derived PDUs), except when a shortage of space in the record route option field causes termination of recording of route, as described above. When Complete Route Recording is selected, PDU reassembly at intermediate systems is performed only when the Derived PDUs that are reassembled all took the same route; otherwise, the PDU is discarded, and if selected, an Error Report is generated (see Clause 6.10).

The second form is referred to as Partial Route Recording. It also requires a record of intermediate systems visited by a PDU. When Partial Route Recording is selected, PDU reassembly at intermediate systems is always permitted. When reassembly is performed at an intermediate system, the route recorded in any of the Derived PDUs may be placed in the PDU resulting from the reassembly.

Note:

The Record Route function is intended to be used in the diagnosis of subnetwork problems and/or to provide a return path that could be used as a source route in a subsequent PDU.

6.16 Quality of Service Maintenance Function

The Quality of Service Maintenance function provides information to network-entities in intermediate systems which may be used to make routing decisions where such decisions affect the overall QoS provided to NS users. This information is conveyed to intermediate system network- entities in a parameter in the options part of the PDU header.

In those instances where the QoS requested cannot be maintained, intermediate system network-entities shall attempt to deliver the PDU at a QoS different from the QoS requested. Intermediate system network-entities do not necessarily provide a notification of failure to meet the requested Quality of Service.

6.17 Priority Function

The Priority function allows a PDU with a numerically higher priority value to be processed preferentially with respect to other PDUs with numerically lower priority values. The function is realized through selection of a parameter in the options part of the PDU header.

The lowest priority value is zero; a source network-entity that does not support the Priority function must set the Priority value to zero. The Priority function provides a means whereby the resources of end and intermediate system network-entities, such as outgoing transmission queues and buffers, can be used preferentially to process higher-priority PDUs ahead of lower-priority PDUs. The specific action taken by an individual network-entity to support the Priority function is a local matter.

6.18 Congestion Notification Function

To allow NS Users to take appropriate action when congestion is experienced within the NS provider, intermediate systems may inform the destination network-entity of congestion through the use of a flag in the QoS Maintenance parameter in the options part of the PDU header. The value of this flag is initially set to zero (0) by the originator of the PDU and may be set to one (1) by any intermediate system which processes the PDU to indicate that it is experiencing congestion. The criteria for determining when this action is to be taken are a local matter.

Note:

Congestion typically corresponds to inavailability of buffer space to maintain output queues. An appropriate policy for indicating congestion may be based upon the depth of the output queue selected for a PDU (according to its destination address or other routing information). When the depth of a particular output queue exceeds a certain proportion of the depth of that queue, an intermediate system will start to discard PDUs. The intermediate system will set the Congestion Experienced flag in the next PDU to be forwarded and may continue to do so until the condition is alleviated.

6.19 Classification of Functions

Implementations are not required to support all of the functions described in Clauses 6.1 through 6.18. Functions are divided into three categories:

Type 1: These functions must be supported.

Type 2: These functions may or may not be supported.
If an implementation does not support a Type 2 function, and the function is selected in a PDU, then that PDU must be discarded, and an Error Report PDU must be generated and forwarded to the originating network-entity, providing that the Error Report flag is set and the conditions of Clause 6.10.4 are satisfied.

Type 3: These functions may or may not be supported.
If an implementation does not support a Type 3 function, and the function is selected in a PDU, then the function is not performed, and the PDU is processed exactly as though the function had not been selected. The protocol data unit shall not be discarded for this reason.

Table 4 shows how the functions are divided into these three categories:

FUNCTION	FULL PROTOCOL	NON SEGMENTING SUBSET	INACTIVE SUBSET
PDU Composition	1	1	1
PDU Composition	1	1	1
Header Format Analysis	1	1	1
PDU Lifetime Control	1	1	N/A
Route PDU	1	1	N/A
Forward PDU	1	1	N/A
Segment PDU	1	N/A	N/A
Reassemble PDU	1	N/A	N/A
Discard PDU	1	1	N/A
Error Reporting (Note 1)	1	1	N/A
Header Error Detection (Note 1)	1	1	N/A
Security	1	2	N/A
Complete Source Routing	1	2	N/A
Complete Route Recording	2	2	N/A
Partial Source Routing	3	3	N/A
Partial Route Recording	3	3	N/A
Priority	3	3	N/A
QoS Maintenance	3	3	N/A
Congestion Notification	3	3	N/A
Padding	3	3	N/A

Table 4: Categorization of Protocol Functions

Note:

1. While the Error Reporting and Header Error Detection functions must be provided, they are provided only when selected by the sending Network Service user.

2. The rationale for the inclusion of type 3 functions is that in the case of some functions it is more important to forward the PDUs between intermediate systems or deliver them to an end-system than it is to support the functions. Type 3 functions should be used in those cases where they are of an advisory nature; they cannot cause a PDU to be discarded when they are not supported.

7 Structure and Encoding of PDUs

7.1 Structure

All Protocol Data Units shall contain an integral number of octets. The octets in a PDU are numbered starting from one (1) and increasing in the order in which they are submitted to the underlying service. The bits in an octet are numbered from one (1) to eight (8), where bit one (1) is the low-order (least significant) bit.

When consecutive octets are used to represent a binary number, the lower octet number has the most significant value.

Any implementation supporting this protocol is required to state in its specification the way in which octets are transferred, using the terms "most significant bit" and "least significant bit". The PDUs of this protocol are defined using the terms "most significant bit" and "least significant bit".

Note:

When the encoding of a PDU is represented using a diagram in this Clause the following representation is used:

a) octets are shown with the lowest numbered octet to the left, higher number octets being further to the right;
b) within an octet, bits are shown with bit eight (8) to the left and bit one (1) to the right.

PDUs shall contain, in the following order:

1. the fixed part;
2. the address part;
3. the segmentation part, if present;
4. the Options part, if present;

and the data field, if present. This structure is illustrated in Figure 2:

7.2 Fixed Part

7.2.1 General

The fixed part of the PDU header contains frequently occurring parameters including the type code (DT or ER) of the protocol data unit. The length and the structure of the fixed part are defined by the PDU code.

The fixed part has the following format:

Part	Described in
Fixed Part	Section 7.2
Address Part	Section 7.3
Segmentation Part	Section 7.4
Options Part	Section 7.5
Data	Section 7.6

Figure 2: PDU Structure

	Octet
Network Layer Protocol Identifier	1
Length Indicator	2
Version/Protocol Id Extension	3
Lifetime	4
SP vline M S vline e/R | Type	5
Segment Length	6,7
Checksum	8,9

Figure 3: PDU Header -- Fixed Part

7.2.2 Network Layer Protocol Identifier

The value of this field is set to binary 1000 0001 to identify this Network Layer protocol as ISO 8473, Protocol for Providing the Connectionless- mode Network Service. The value of this field is set

to binary 0000 0000 to identify the Inactive Network Layer protocol subset.

7.2.3 Length Indicator

The length is indicated by a binary number, with a maximum value of 254 (1111 1110). The length indicated is the length in octets of the header, as described in Clause 7.1. The value 255 (1111 1111) is reserved for possible future extensions.

Note:

The rules for forwarding and segmentation guarantee that the header length is the same for all segments (Derived PDUs) of the Initial PDU, and is the same as the header length of the Initial PDU. The size of a PDU header will not change due to operation of any protocol function.

7.2.4 Version/Protocol Identifier Extension

The value of this field is binary 0000 0001, which identifies the standard Version 1 of ISO 8473, Protocol for Providing the Connectionless-mode Network Service.

7.2.5 PDU Lifetime

The PDU Lifetime field is encoded as a binary number representing the remaining lifetime of the PDU, in units of 500 milliseconds.

7.2.6 Flags

7.2.6.1 Segmentation Permitted

The Segmentation Permitted flag indicates whether segmentation is permitted. Its value is determined by the originator of the PDU and cannot be changed by any other network-entity for the lifetime of the Initial PDU and any Derived PDUs.

A value of one (1) indicates that segmentation is permitted. A value of zero (0) indicates that the non-segmenting protocol subset is employed. When the value of zero is selected, the segmentation part of the PDU header is not present, and the Segment Length field serves as the Total Length field (see Clause 7.2.8).

7.2.6.2 More Segments

The More Segments flag indicates whether the data segment in this PDU contains (as its last octet) the last octet of the User Data in the NSDU. When the More Segments flag is set to one (1), segmentation has taken place and the last octet of the NSDU is not contained in this PDU. The More Segments flag cannot be set to one (1) if the Segmentation Permitted flag is not set to one (1).

When the More Segments flag is set to zero (0), the last octet of the Data Part of the PDU is the last octet of the NSDU.

7.2.6.3 Error Report

When the Error Report flag is set to one, the rules in Clause 6.10 are used to determine whether to generate an Error Report PDU if it is necessary to discard this Data PDU.

When the Error Report flag is set to zero, discard of the Data PDU will not cause the generation of an Error Report PDU.

7.2.7 Type Code

The Type code field identifies the type of the protocol data unit. Allowed values are given in Table 5:

	Bits	5	4	3	2	1
DT PDU		1	1	1	0	0
ER PDU		0	0	0	0	1

Table 5: Valid PDU Types

7.2.8 PDU Segment Length

The Segment Length field specifies the entire length, in octets, of the Derived PDU, including both header and data (if present). When the full protocol is employed and a PDU is not segmented, the value of this field is identical to the value of the Total Length field located in the Segmentation Part of the header.

When the non-segmenting protocol subset is employed, no segmentation part is present in the header. In this subset, the Segment Length field specifies the entire length of the Initial PDU, including both header and data (if present). The Segment Length field is not changed for the lifetime of the PDU.

7.2.9 PDU Checksum

The checksum is computed on the entire PDU header. For the Data PDU, this includes the segmentation and options parts (if present). For the Error Report PDU, this includes the reason for discard field as well.

A checksum value of zero is reserved to indicate that the checksum is to be ignored. The operation of the PDU Header Error Detection function (Clause 6.11) ensures that the value zero does not represent a

valid checksum. A non-zero value indicates that the checksum must be processed. If the checksum calculation fails, the PDU must be discarded.

7.3 Address Part

7.3.1 General

Address parameters are distinguished by their location, immediately following the fixed part of the PDU header. The address part is illustrated Figure 4:

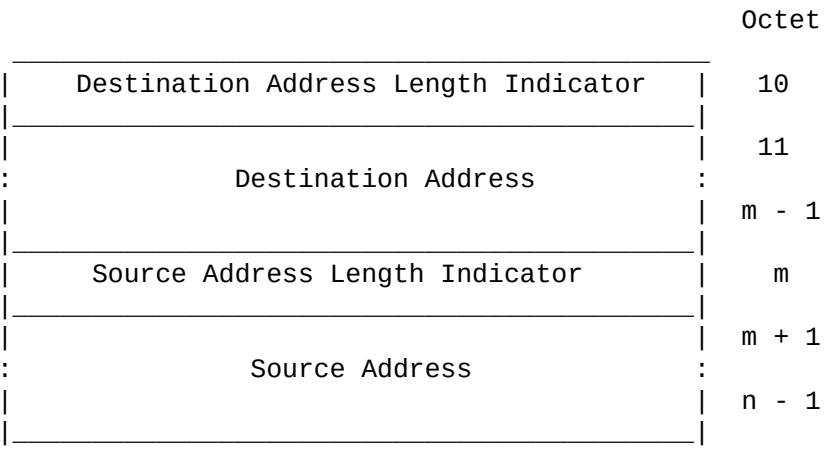

Figure 4: PDU Header -- Address Part

7.3.1.1 Destination and Source Addresses

The Destination and Source addresses used by this protocol are Network Service Access Point addresses as defined in ISO 8348/AD2, Addendum to the Network Service Definition Covering Network Layer Addressing.

The Destination and Source Addresses are variable length. The Destination and Source Address fields are encoded as Network Protocol Address Information using the Preferred Binary Encoding defined in Clause 8.3.1 of ISO 8348/AD2.

The Destination Address Length Indicator field specifies the length of the Destination Address in octets. The Destination Address field follows the Destination Address Length Indicator field.

The Source Address Length Indicator field specifies the length of the Source Address in octets. The Source Address Length Indicator field follows the Destination Address field. The Source Address field follows the Source Address Length Indicator field.

Each address parameter is encoded as illustrated in Table 5:

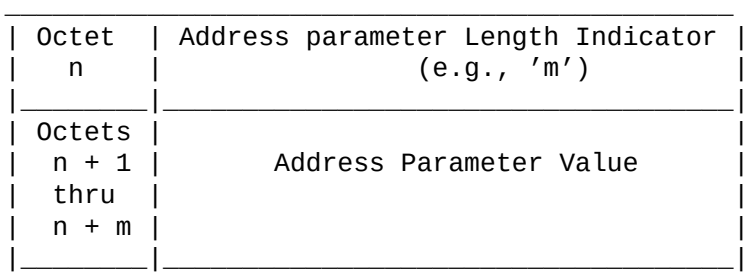

Figure 5: Address Parameters

7.4 Segmentation Part

If the Segmentation Permitted Flag in the Fixed Part of the PDU Header (Octet 4, Bit 8) is set to one, the segmentation part of the header, illustrated in Figure 6, must be present:

If the Segmentation Permitted flag is set to zero, the non-segmenting protocol subset is in use.

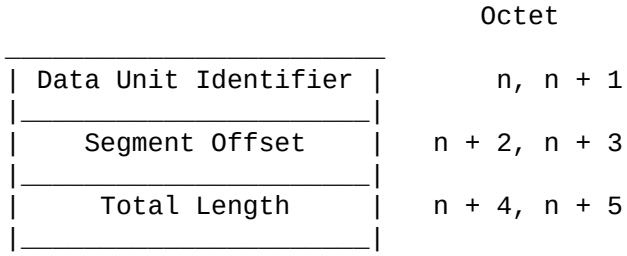

Figure 6: PDU Header -- Segmentation Part

7.4.1 Data Unit Identifier

The Data Unit Identifier identifies an Initial PDU (and hence, its Derived PDUs) so that a segmented data unit may be correctly reassembled. The Data Unit Identifier size is two octets.

7.4.2 Segment Offset

For each Derived PDU, the Segment Offset field specifies the relative position of the segment contained in the data field of the Derived PDU with respect to the start of the data field of the Initial PDU. The offset is measured in units of octets. The offset of the first segment (and hence, the Initial PDU) is zero; an unsegmented (Initial PDU) has a segment offset value of zero (0). The value of this field shall be a multiple of eight 8).

7.4.3 PDU Total Length

The Total Length field specifies the entire length of the Initial PDU, including both the header and data. This field is not changed for the lifetime of the Initial PDU (and hence, its Derived PDUs).

7.5 Options Part

7.5.1 General

The options part is used to convey optional parameters. The options part of the PDU header is illustrated below:

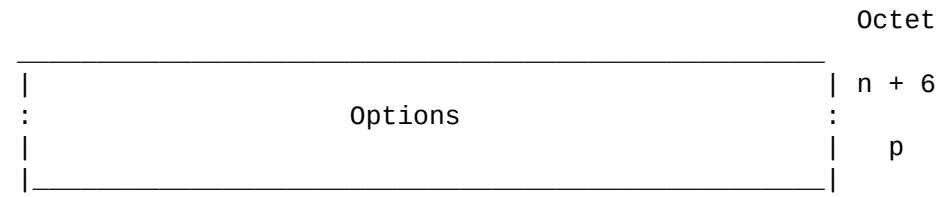

Figure 7: PDU Header -- Options Part

If the options part is present, it may contain one or more parameters. The number of parameters that may be contained in the options part is constrained by the length of the options part, which is determined by the following formula:

PDU Header Length -(length of fixed part+length of address part+length of segmentation part)

and by the length of the individual optional parameters.

Parameters defined in the options part may appear in any order. Duplication of options is not permitted. Receipt of a Protocol Data Unit with an option duplicated should be treated as a protocol error. The rules governing the treatment of protocol errors are described in Clause 6.10, Error Reporting Function.

The encoding of parameters contained within the options part of the PDU header is illustrated in Table 6:

Octets	
n	Parameter Code
n + 1	Parameter Length (e.g.m)
n + 2 to n + m + 1	Parameter Value

Table 6: Encoding of Parameters

The parameter code field is coded in binary and, without extensions, provides a maximum of 255 different parameters. No parameter codes use bits 8 and 7 with the value 00, so the actual maximum number of parameters is lower. A parameter code of 255 (binary 1111 1111) is reserved for possible future extensions.

The parameter length field indicates the length, in octets, of the parameter value field. The length is indicated by a positive binary number, m, with a theoretical maximum value of 254. The practical maximum value of m is lower. For example, in the case of a single parameter contained within the options part, two octets are required for the parameter code and the parameter length indicators. Thus, the value of m is limited to:

m = 252-(length of fixed part +length of address part +length of segmentation part)

For each succeeding parameter the maximum value of m decreases. The parameter value field contains the value of the parameter identified in the parameter code field.

The following parameters are permitted in the options part.

7.5.2 Padding

The padding parameter is used to lengthen the PDU header to a convenient size (See Clause 6.12).

Parameter Code: 1100 1100

Parameter Length: variable

Parameter Value: any value is allowed

7.5.3 Security

This parameter allows a unique and unambiguous security level to be assigned to a protocol data unit.

Parameter Code: 1100 0101

Parameter Length: variable

Parameter Value: The high order two bits of the first octet specify the Security Format Code, where:

Security Format Code	Type of Security Field:

00	Reserved
01	Source Address Specific
10	Destination Address Specific
11	Globally Unique

The rest of the first octet is reserved and must be zero. The remainder of the Parameter Value field specifies the security level as described in the following Clauses.

7.5.3.1 Source Address Specific

The Security Format Code value of binary "01" indicates that the remaining octets of the parameter value field specify a security level which is unique and unambiguous in the context of the security classification system employed by the authority responsible for assigning the source NSAP Address.

7.5.3.2 Destination Address Specific

The Security Format Code value of binary "10" indicates that the remaining octets of the parameter value field specify a security level which is unique and unambiguous in the context of the security classification system employed by the authority responsible for assigning the destination NSAP Address.

7.5.3.3 Globally Unique Security

The Security Format Code value of binary "11" indicates that the remaining octets of the parameter value field specify a globally unique and unambiguous security level. This security classification system is not specified in this Standard.

7.5.4 Source Routing

The source routing parameter specifies, either completely or partially, the route to be taken from Source Network Address to Destination Network Address.

Parameter Code: 1100 0101

Parameter Length: variable

Parameter Value: 2 octets of control information succeeded by a concatenation of ordered network-entity title entries (ordered from source to destination)

The first octet of the parameter value is the type code, and has the following significance:

0000 0000	partial source routing
0000 0001	complete source routing
	<all other values reserved>

The second octet indicates the octet offset of the next network-entity title entry to be processed in the list. It is relative to the start of the parameter, such that a value of three (3) indicates that the next network-entity title entry begins immediately after this control octet. Successive octets are indicated by correspondingly larger values of this indicator.

The third octet begins the network-entity title list. The list consists of variable length network-entity title entries. The first octet of entry identifies the length of the network-entity title which comprises the re- mainder of the entry.

7.5.5 Recording of Route

The recording of route parameter identifies the route of intermediate systems traversed by the PDU.

Parameter Code: 1100 1011

Parameter Length: variable

Parameter Value: 2 octets of control information succeeded by a con catenation of ordered network-entity title entries (ordered from destination to source)

The first octet of the parameter value is the type code, and has the following significance:

```
0000 0000    Partial Recording of Route in progress
0000 0001    Complete Recording of Route in progress
             <all other values reserved>
```

The second octet identifies the first octet not currently used for a recorded network-entity title, and therefore also the end of the list. It is encoded relative to the start of the parameter value, such that a value of three (3) indicates that no network-entity titles have yet been recorded. A value of all ones is used to indicate that route recording has been terminated.

The third octet begins the network-entity title list. The list consists of variable length network-entity title entries. The first octet of each entry specifies the length of the network-entity title comprising the remainder of the entry. Network-entity title entries are always added to the beginning of the list; that is, the most recently added entry will begin in the third octet of the parameter value.

Note:

The length of the Record Route parameter is determined by the originator of the PDU and is not changed during the lifetime of the PDU; hence, the operation of the Record Route function does

not affect the length of the header.

7.5.6 Quality of Service Maintenance

The Quality of Service parameter conveys information about the quality of service requested by the originating Network Service user. Network-entities in intermediate systems may (but are not required to) make use of this information as an aid in selecting a route when more than one route satisfying other routing criteria is available and the available routes are known to differ with respect to Quality of Service see Clause 6.16).

Parameter Code: 1100 0011
Parameter Length: variable
Parameter Value: The high order two bits of the first octet specify the QoS Format Code, where:

QoS Format Code	Type of QoS Field
00	Reserved
01	Source Address Specific
10	Destination Address Specific
11	Globally Unique

The rest of the first octet is reserved and must be zero. The remainder of the Parameter Value field specifies the QoS as described in the following Clauses.

7.5.6.1 Source Address Specific

The QoS Format Code value of binary "01" indicates that the remaining octets of the parameter value field specify a QoS which is unique and unambiguous in the context of the QoS Maintenance system employed by the authority responsible for assigning the source NSAP Address.

7.5.6.2 Destination Address Specific

The QoS Format Code value of binary "10" indicates that the remaining octets of the parameter value field specify a QoS which is unique and unambiguous in the context of the QoS Maintenance system employed by the authority responsible for assigning the destination NSAP Address.

7.5.6.3 Globally Unique QoS

The QoS Format Code value of binary "11" indicates that the remainder of the parameter value field specifies a globally unique QoS Maintenance field. When the globally unique QoS Maintenance function is employed, the parameter value field must have a total length of one octet, which is assigned the following values:

Bits 8 and 7: QoS Format Code of binary "11"

```
Bit 6:          Reserved
Bit 5:          sequencing vs. transit delay
Bit 4:          congestion experienced
Bit 3:          transit delay vs. cost
Bit 2:          residual error probability vs. transit delay
Bit 1:          residual error probability vs. cost
```

Bit 5 is set to one to indicate that, where possible, routing decisions should favor sending all PDUs to the specified destination NSAP address over a single path (in order to maintain sequence) over minimizing transit delay. A value of zero (0) indicates that, where possible, routing decisions should favor low transit delay over sequence preservation.

Bit 4 is set to zero by the network-entity which originates the protocol data unit. It is set to one by an intermiediate system to indicate that this PDU has visited a congested intermediate system, and appropriate action should be taken by the destination network-entity. Once the congestion experienced bit is set by an intermediate system, it may not be reset by any intermediate system traversed by the PDU further along the path towards the destination.

Bit 3 is set to one to indicate that, where possible, routing decisions should favor low transit delay over low cost. A value of 0 indicates that routing decisions should favor low cost over low transit delay.

Bit 2 set to one to indicate that, where possible, routing decisions should favor low residual error probability over low transit delay. A value of zero indicates that routing decisions should favor low transit delay over low residual error probability.

Bit 1 is set to one to indicate that, where possible, routing decisions should favor low residual error probability over low cost. A value of 0 indicates that routing decisions should favor low cost over low residual error probability.

7.5.7 Priority

The value of the Priority parameter indicates the relative priority of the protocol data unit. Intermediate systems that support this option shall make use of this information in routing and in ordering PDUs for transmission.

```
Parameter Code:        1100 1101

Parameter Length:      one octet

Parameter Value:       0000 0000 - Normal (Default) through
       0000 1110 - Highest
       <all other values reserved>
```

The values 0000 0001 through 0000 1110 are to be used for higher priority protocol data units. If an intermediate system does not support this option, all PDUs shall be treated as if the field had the value 0000 0000.

7.6 Data Part

The Data part of the PDU is structured as an ordered multiple of octets, which is identical to the same ordered multiple of octets specified for the NS-Userdata parameter of the N-UNITDATA Request and Indication primitives. The data field is illustrated in Figure 8:

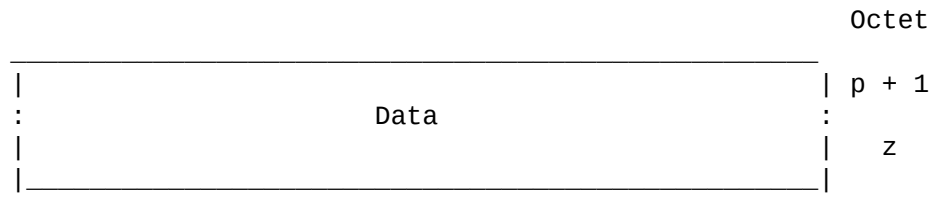

Figure 8: PDU Header -- Data Field

7.7 Data (DT) PDU

7.7.1 Structure

The DT PDU has the following format:

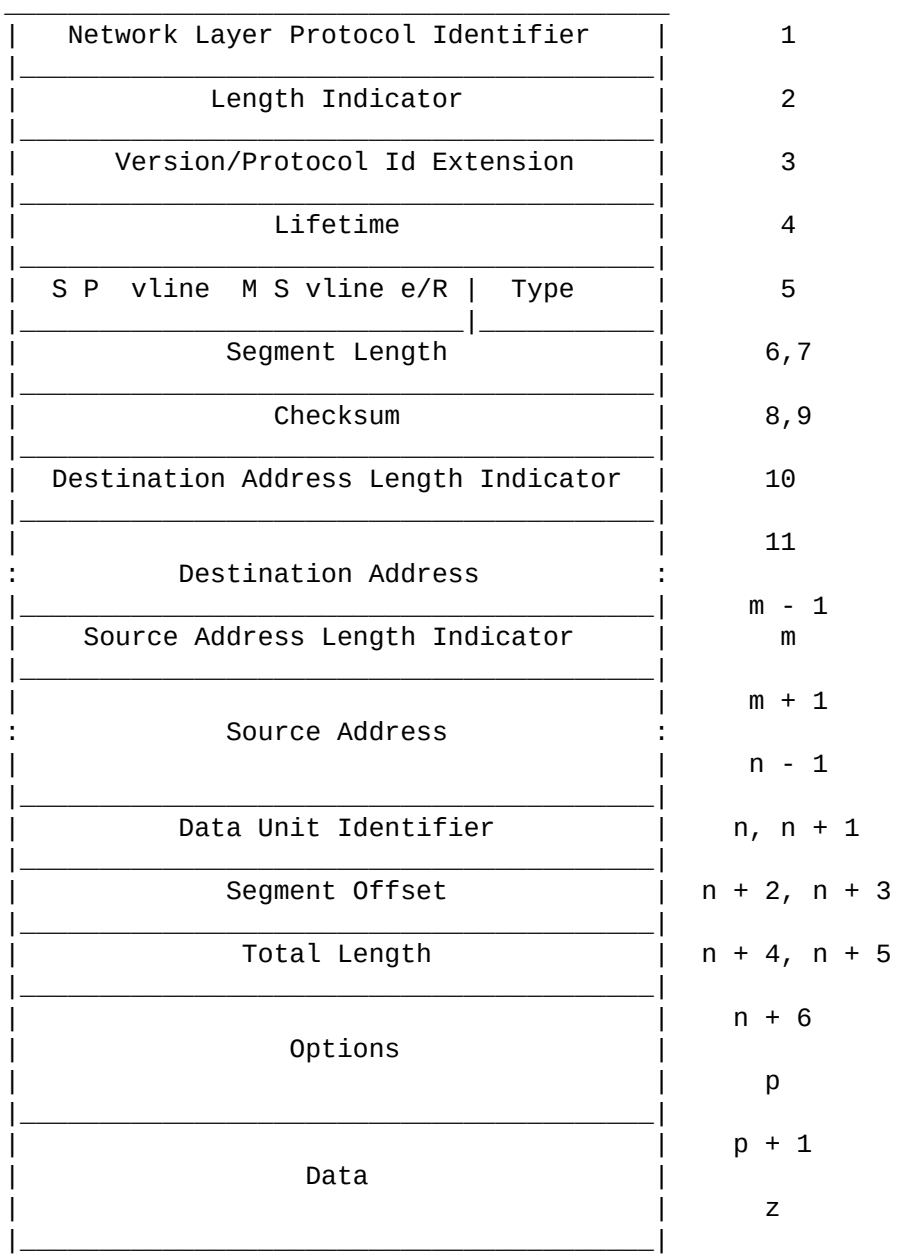

Figure 9: DT PDU

7.7.1.1 Fixed Part

1)	Network Layer Protocol Identifier	See Clause 7.2.2
2)	Length Indicator	See Clause 7.2.3
3)	Version/Protocol Id Extension	See Clause 7.2.4
4)	Lifetime	See Clause 7.2.5
5)	SP, MS, E/R	See Clause 7.2.6
6)	Type Code	See Clause 7.2.7
7)	Segment Length	See Clause 7.2.8
8)	Checksum	See Clause 7.2.9

7.7.1.2 Addresses

See Clause 7.3.

7.7.1.3 Segmentation

See Clause 7.4.

7.7.1.4 Options

See Clause 7.5.

7.7.1.5 Data

See Clause 7.7.

7.8 Inactive Network Layer Protocol

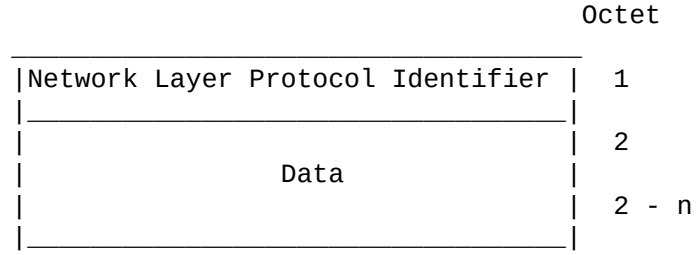

Figure 10: Inactive Network Layer Protocol

7.8.1 Network Layer Protocol Id

The value of the Network Layer Protocol Identifier field is binary zero (0000 0000).

7.8.2 Data Field

The length of the NS-Userdata parameter is constrained to be less than or equal to the value of the length of the SN-Userdata parameter minus one (see Clause 7.7).

7.9 Error Report PDU (ER)

7.9.1 Structure

The ER PDU has the following format:

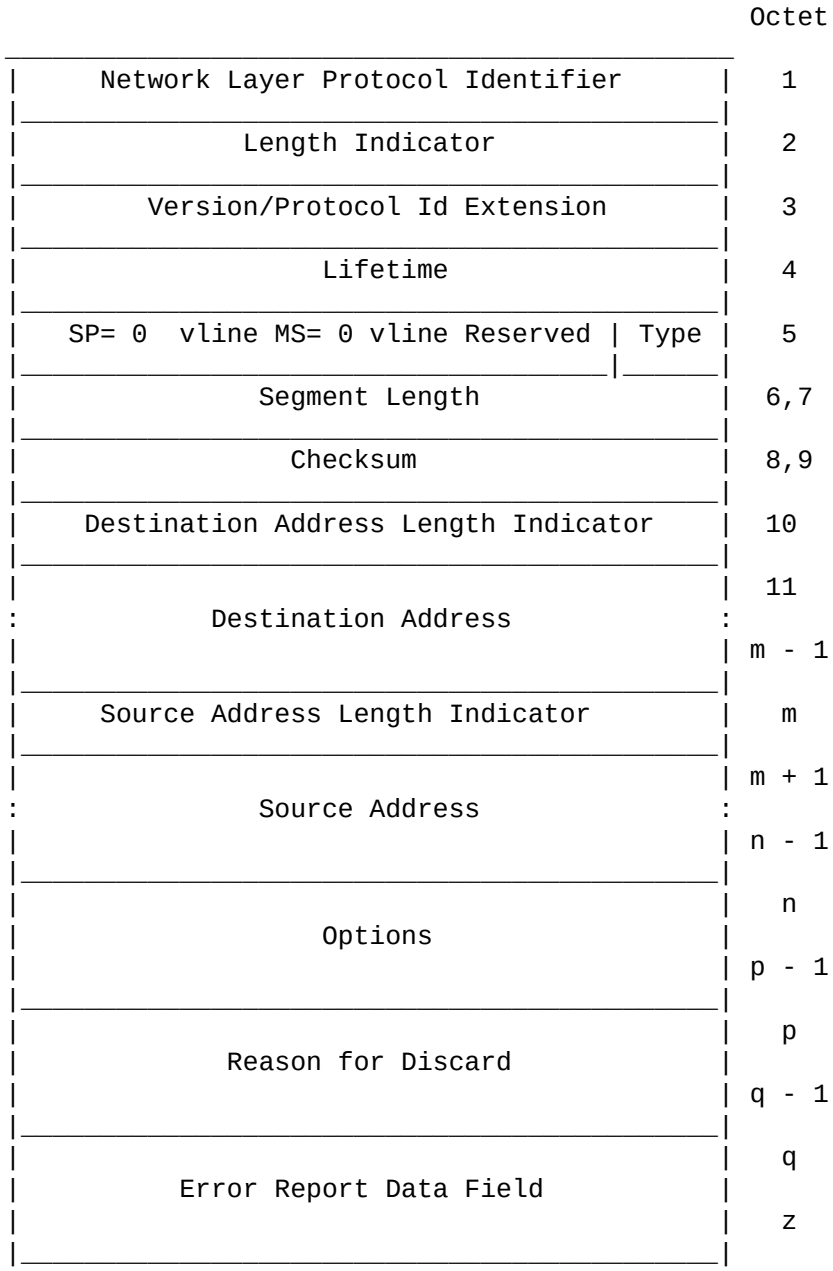

Figure 11: Error Report PDU

7.9.1.1 Fixed Part

The fixed part of the Error Report Protocol Data Unit is composed in the same way as a new (Initial) Data PDU. References are provided to previous Clauses describing the encoding of the fields comprising the fixed part:

1)	Network Layer Protocol Identifier	See Clause 7.2.2
2)	Length Indicator	See Clause 7.2.3
3)	Version/Protocol Id Extension	See Clause 7.2.4
4)	Lifetime	See Clause 7.2.5
5)	SP, MS, E/R	Always set to zero, (See Clause 6.10)
6)	Type Code	See Clause 7.2.7
7)	Segment Length	See Clause 7.2.8
8)	Checksum	See Clause 7.2.9

7.9.1.2 Addresses

See Clause 7.3.

The Destination Address specifies the network-entity title of the originator of the discarded PDU. The Source Address specifies the title of the intermediate-system or end-system network-entity initiating the Error Report PDU.

7.9.1.3 Options

See Clause 7.5.

7.9.1.4 Reason for Discard

This parameter is valid only for the Error Report PDU.

Parameter Code: 1100 0001
Parameter Length: two octets
Parameter Value: type of error encoded in binary. Values are listed in Table 7:

Parameter Value Octet 1	Octet 2	Class of Error	Meaning
0000	0000		Reason not specified
	0001		Protocol Procedure Error
	0010		Incorrect Checksum
	0011	General	PDU Discarded due to Congestion
	0100		Header Syntax Error (cannot be parsed)
	0101		Segmentation needed but not permitted
	0110		Incomplete PDU Received
	0111		Duplicate Option
1000	0000	Address	Destination Address Unreachable
	0001		Destination Address Unknown
1001	0000		Unspecified Source Routing Error
	0001	Source	Syntax Error in Source Routing Field
	0010	Routing	Unknown Address in Source Routing Field
	0011		Path not Acceptable
1010	0000	Lifetime	Lifetime Expired while Data Unit in Transit
	0001		Lifetime Expired during Reassembly
1011	0000		Unsupported Option not Specified
	0001	PDU	Unsupported Protocol Version
	0010	Discarded	Unsupported Security Option
	0011		Unsupported Source Routing Option
	0100		Unsupported Recording of Route Option
1100	0000	Reassembly	Reassembly interference

Table 7: Reasons for Discard

The first octet of the parameter value contains an error type code. If the error in the discarded Data PDU can be localized to a particular field, the number of the first octet of that field is stored in the second octet of the reason for discard parameter field. If the error cannot be localized to a particular field, or if the error is a checksum error, then the value zero is stored in the second octet of the reason for discard parameter field.

7.9.1.5 Error Report Data Field

This field contains the entire header of the discarded Data PDU, and may contain some or all of the data field of the discarded PDU.

8 Conformance

For conformance to this International Standard, the ability to originate, manipulate, and receive PDUs in accordance with the full protocol (as opposed to the non-segmenting or Inactive Network Layer Protocol subsets) is required.

Additionally, conformance to the Standard requires provision of the protocol functions described in Clause 6. Provision of the optional functions described in Clause 6.18 and enumerated in Table 9-1 must meet the requirements described therein. Exceptions to this requirement are described in Clause 8.1 below.

Additionally, conformance to the Standard requires adherence to the structure and encoding of PDUs of Clause 7.

If and only if the above requirements are met is there conformance to this International Standard.

8.1 Provision of Functions for Conformance

The following table categorizes the functions in Clause 6 with respect to the type of system providing the function:

Note:

1. The support of the PDU Composition and Forward PDU functions is necessary for the generation of Error Report PDUs.

2. The Segment PDU function is in general mandatory for an intermediate system. However, a system which is to be connected only to subnetworks all offering the same maximum SDU size (such as identical Local Area Networks) will not need to perform this function and therefore does not need to implement it.

   If this function is not implemented, this shall be stated as part of the specification of the implementation.

3. The correct treatment of the padding function requires no processing. A conforming implementation shall support the function, to the extent of ignoring this parameter wherever it may appear.

4. This function may or may not be supported. If an implementation does not support this function, and the

function is selected in a PDU, then the PDU shall be discarded, and an ER PDU shall be generated and forwarded to the originating network-entity if the Error Report flag is set and the conditions of Clause 6.10.4 are satisfied.

5. This function may or may not be supported. If an implementation does not support this function, and the function is selected in a PDU, then the function is not performed and the PDU is processed exactly as though the function had not been selected. The PDU shall not be discarded for this reason.

Function	Send	Forward	Receive
PDU Composition	M	(Note 1)	(Note 1)
PDU Decomposition	M	-	M
Header Format Analysis	-	M	M
PDU Lifetime Control		M	I
Route PDU	-	M	-
Forward PDU	M	M	(Note 1)
Segment PDU	M	(Note 2)	-
Reassemble PDU	-	I	M
Discard PDU	-	M	M
Error Reporting	M	M	M
Header Error Detection	(Note 3)	M	M
Security	-	(Note 3)	(Note 4)
Complete Source Routing	-	(Note 4)	-
Complete Route Recording	-	(Note 4)	-
Partial Source Routing	-	(Note 5)	-
Partial Route Recording	-	(Note 5)	-
Priority	-	(Note 5)	-
QoS Maintenance	-	(Note 5)	-
Congestion Notification	-	(Note 5)	-
Padding	-	(Note 5)	(Note 3)

Table 8: Categorization of Functions

Key:

M: Mandatory Function; this function must be implemented.

-: Not applicable.

I: Implementation option, as described in the text.

NOTE: See notes above