

AS/NZS 2777.1:1998

ISO/IEC 7498-1:1994

(Corrected & reprinted 1996)

Australian/New Zealand Standard[®]

**Information processing systems—
Open Systems Interconnection—
Basic reference model**

Part 1: The basic model

[ISO/IEC title: Information technology—Open Systems
Interconnection—Basic Reference Model: The Basic Model]

STANDARDS AUSTRALIA ✓

STANDARDS NEW ZEALAND



Documents

Document Sharing Hub

AS/NZS 2777.1:1998

This Joint Australian/New Zealand Standard was prepared by Joint Technical Committee IT/1, Information Systems—Interconnection. It was approved on behalf of the Council of Standards Australia on 29 October 1997 and on behalf of the Council of Standards New Zealand on 16 December 1997. It was published on 5 February 1998.

The following interests are represented on Committee IT/1:

Australian Bankers Association
 Australian Bureau of Statistics
 Australian Computer Society
 Australian Computer Users Association
 Australian Information Industry Association
 Australian Vice Chancellors Committee
 Department of Defence, Australia
 Department of Industry, Science and Technology, Australia
 Government Computing Service, New Zealand
 Information Exchange Steering Committee, Australia
 Institute of Information and Communication Technologies, CSIRO, Australia
 Telstra Corporation, Australia
 Telecom New Zealand

Review of Standards. To keep abreast of progress in industry, Joint Australian/New Zealand Standards are subject to periodic review and are kept up to date by the issue of amendments or new editions as necessary. It is important therefore that Standards users ensure that they are in possession of the latest edition, and any amendments thereto.

Full details of all Joint Standards and related publications will be found in the Standards Australia and Standards New Zealand Catalogue of Publications; this information is supplemented each month by the magazines 'The Australian Standard' and 'Standards New Zealand', which subscribing members receive, and which give details of new publications, new editions and amendments, and of withdrawn Standards.

Suggestions for improvements to Joint Standards, addressed to the head office of either Standards Australia or Standards New Zealand, are welcomed. Notification of any inaccuracy or ambiguity found in a Joint Australian/New Zealand Standard should be made without delay in order that the matter may be investigated and appropriate action taken.

AS/NZS 2777.1:1998

Australian/New Zealand Standard®

**Information processing systems—
Open Systems Interconnection—
Basic reference model**

Part 1: The basic model

Originated in Australia as AS 2777 — 1985.
Jointly revised and designated as AS/NZS 2777.1:1996.
Second edition 1998.

PUBLISHED JOINTLY BY:

STANDARDS AUSTRALIA
1 The Crescent,
Homebush NSW 2140 Australia

STANDARDS NEW ZEALAND
Level 10, Radio New Zealand House,
155 The Terrace,
Wellington 6001 New Zealand

PREFACE

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT/1, Information Systems—Interconnection. It is identical with and has been reproduced from ISO/IEC 7498-1:1994 (corrected and reprinted 1996), *Information technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*, and supersedes AS/NZS 2777.1:1996, *Information processing systems—Open Systems Interconnection—Basic reference model, Part 1: The basic model*.

The objective of this Standard is to provide a common basis for the coordination of Standards development for the purpose of systems interconnection, while allowing existing Standards to be placed in perspective within the overall Reference Model.

This Standard is one of a series of Open Systems Interconnection (OSI) Standards which are currently under development. Since OSI Standards are developmental, there may be some minor difficulties encountered in their implementation. For this reason, Standards Australia and Standards New Zealand will be providing, through the OSI Help Desk, a service to coordinate and disseminate information concerning difficulties which are identified in using this Standard.

Users of this Standard are advised by Standards Australia and Standards New Zealand, under arrangements made with ISO and IEC, as well as certain other Standards organizations, that the number of this Standard is not reproduced on each page; its identity is shown only on the cover and title pages.

© Copyright – STANDARDS AUSTRALIA/STANDARDS NEW ZEALAND

Users of Standards are reminded that copyright subsists in all Standards Australia and Standards New Zealand publications and software. Except where the Copyright Act allows and except where provided for below no publications or software produced by Standards Australia or Standards New Zealand may be reproduced, stored in a retrieval system in any form or transmitted by any means without prior permission in writing from Standards Australia or Standards New Zealand. Permission may be conditional on an appropriate royalty payment. Australian requests for permission and information on commercial software royalties should be directed to the head office of Standards Australia. New Zealand requests should be directed to Standards New Zealand.

Up to 10 percent of the technical content pages of a Standard may be copied for use exclusively in-house by purchasers of the Standard without payment of a royalty or advice to Standards Australia or Standards New Zealand.

Inclusion of copyright material in computer software programs is also permitted without royalty payment provided such programs are used exclusively in-house by the creators of the programs.

Care should be taken to ensure that material used is from the current edition of the Standard and that it is updated whenever the Standard is amended or revised. The number and date of the Standard should therefore be clearly identified.

The use of material in print form or in computer software programs to be used commercially, with or without payment, or in commercial contracts is subject to the payment of a royalty. This policy may be varied by Standards Australia or Standards New Zealand at any time.

CONTENTS

	<i>Page</i>
1 Scope.....	1
2 Definitions.....	2
3 Notation.....	2
4 Introduction to Open Systems Interconnection (OSI).....	2
4.1 Definitions.....	2
4.2 Open System Interconnection Environment	3
4.3 Modelling the OSI Environment.....	4
5 Concepts of a layered architecture	5
5.1 Introduction.....	5
5.2 Principles of layering	6
5.3 Communication between peer-entities.....	9
5.4 Identifiers	13
5.5 Properties of service-access-points	14
5.6 Data-units	15
5.7 The nature of the (N)-service.....	16
5.8 Elements of layer operation	16
5.9 Routing.....	27
5.10 Quality Of Service (QOS).....	27
6 Introduction to the specific OSI layers.....	28
6.1 Specific layers	28
6.2 The principles used to determine the seven layers in the Reference Model	29
6.3 Layer descriptions.....	30
6.4 Combinations of connection-mode and connectionless-mode.....	30
6.5 Configurations of OSI Open Systems.....	31
7 Detailed description of the resulting OSI architecture	32
7.1 Application Layer	32
7.2 Presentation Layer	33
7.3 Session Layer	34
7.4 Transport Layer.....	37
7.5 Network Layer	41
7.6 Data Link Layer	46
7.7 Physical Layer.....	49
8 Management aspects of OSI.....	52
8.1 Definitions.....	52
8.2 Introduction.....	53
8.3 Categories of management activities	53
8.4 Principles for positioning management functions.....	54
9 Compliance and Consistency with this reference model.....	54
9.1 Definitions.....	54
9.2 Application of consistency and compliance requirements.....	55
Annex A – Brief explanation of how the layers were chosen	56
Annex B – Alphabetical index to definitions	57

AUSTRALIAN/NEW ZEALAND STANDARD

Information processing systems—Open Systems Interconnection— Basic reference model

Part 1: The basic model

1 Scope

1.1 The purpose of this Reference Model of Open Systems Interconnection is to provide a common basis for the coordination of standards development for the purpose of systems interconnection, while allowing existing standards to be placed into perspective within the overall Reference Model.

1.2 The term Open Systems Interconnection (OSI) qualifies standards for the exchange of information among systems that are “open” to one another for this purpose by virtue of their mutual use of the applicable standards.

1.3 The fact that a system is open does not imply any particular systems implementation, technology or means of interconnection, but refers to the mutual recognition and support of the applicable standards.

1.4 It is also the purpose of this Reference Model to identify areas for developing or improving standards, and to provide a common reference for maintaining consistency of all related standards. It is not the intent of this Reference Model either to serve as an implementation specification, or to be a basis for appraising the conformance of actual implementations, or to provide a sufficient level of detail to define precisely the services and protocols of the interconnection architecture. Rather, this Reference Model provides a conceptual and functional framework which allows international teams of experts to work productively and independently on the development of standards for each layer of the Reference Model for OSI.

1.5 The Reference Model has sufficient flexibility to accommodate advances in technology and expansion in user demands. This flexibility is also intended to allow the phased transition from existing implementations to OSI standards.

1.6 While the scope of the general architectural principles required for OSI is very broad, this Reference Model is primarily concerned with systems comprising terminals, computers, and associated devices and the means for transferring information between such systems. Other aspects of OSI requiring attention are described briefly (see 4.2).

1.7 The description of the Basic Reference Model of OSI is developed in stages:

1.8 Clause 4 establishes the reasons for Open Systems Interconnection, defines what is being connected, the scope of the interconnection, and describes the modelling principles used in OSI.

1.9 Clause 5 describes the general nature of the architecture of the Reference Model; namely that it is layered, what layering means, and the principles used to describe layers.

1.10 Clause 6 names, and introduces the specific layers of the architecture.

1.11 Clause 7 provides the description of the specific layers.

1.12 Clause 8 provides the description of Management Aspects of OSI.

1.13 Clause 9 specifies compliance and consistency with the OSI Reference Model.

1.14 An indication of how the layers were chosen is given in Annex A to this Basic Reference Model.

1.15 Additional aspects of this Reference Model beyond the basic aspects are described in several parts. The first part describes the Basic Reference Model. The second part describes the architecture for OSI Security. The third part describes OSI Naming and Addressing. The fourth describes OSI System Management.

1.16 The Basic Reference Model serves as a framework for the definition of services and protocols which fit within the boundaries established by the Reference Model.

1.17 In those few cases where a feature is explicitly marked (optional) in the Basic Reference Model it should remain optional in the corresponding service or protocol (even if at a given instant the two cases of the option are not yet documented).

1.18 This Reference Model does not specify services and protocols for OSI. It is neither an implementation specification for systems, nor a basis for appraising the conformance of implementations.

1.19 For standards which meet the OSI requirements, a small number of practical subsets are defined from optional functions, to facilitate implementation and compatibility.

2 Definitions

Definitions of terms are included at the beginning of individual clauses and sub-clauses. An index of these terms is provided in Annex B for easy reference.

3 Notation

3.1 Layers are introduced in clause 5. An (N)-, (N+1)- and (N-1)- notation is used to identify and relate adjacent layers:

(N)-layer: any specific layer;

(N+1)-layer: the next higher layer;

(N-1)-layer: the next lower layer.

This notation is also used for other concepts in the model which are related to these layers, for example (N)-protocol, (N+1)-service.

3.2 Clause 6 introduces names for individual layers. When referring to these layers by name, the (N)-, (N+1)- and (N-1)- prefixes are replaced by the names of the layers, for example transport-protocol, session-entity, and network-service.

4 Introduction to Open Systems Interconnection (OSI)

NOTE – The general principles described in clauses 4 and 5 hold for all layers of the Reference Model, unless layer specific statements to the contrary are made in clauses 6 and 7.

4.1 Definitions

4.1.1 real system: A set of one or more computers, the associated software, peripherals, terminals, human operators, physical processes, information transfer means, etc., that forms an autonomous whole capable of performing information processing and/or information transfer.

4.1.2 real open system: A real system which complies with the requirements of OSI standards in its communication with other real systems.

4.1.3 open system: The representation within the Reference Model of those aspects of a real open system that are pertinent to OSI.

4.1.4 application process: An element within a real open system which performs the information processing for a particular application.

4.1.5 Open System Interconnection Environment (OSIE): An abstract representation of the set of concepts, elements, functions, services, protocols, etc., as defined by the OSI Reference Model and the derived specific standards which, when applied, enable communications among open systems.

4.1.6 Local System Environment (LSE): An abstract representation of that part of the real system that is not pertinent to OSI.

NOTE – The LSE may include functions necessary for non-OSI communication.

4.1.7 application-process-invocation: A specific utilization of part or all of the capabilities of a given application process in support of a specific occasion of information processing.

4.1.8 application-process-type: A description of a class of application processes in terms of a set of information processing capabilities.

4.2 Open System Interconnection Environment

4.2.1 In the concept of OSI, a real system is a set of one or more computers, associated software, peripherals, terminals, human operators, physical processes, information transfer means, etc., that forms an autonomous whole capable of performing information processing and/or information transfer.

4.2.2 An application process is an element within a real open system which performs the information processing for a particular application.

4.2.3 Application processes can represent manual processes, computerized processes or physical processes.

4.2.4 Some examples of application processes that are applicable to this open system definition are the following:

- a) a person operating a banking terminal is a manual application-process;
- b) a FORTRAN program executing in a computer center and accessing a remote database is a computerized application-process; the remote database management systems server is also an application-process; and
- c) a process control program executing in a dedicated computer attached to some industrial equipment and linked into a plant control system is a physical application-process.

4.2.5 An application process represents a set of resources, including processing resources, within a real open system that may be used to perform a particular information processing activity. An application process may organize its interactions with other application processes in whatever way is necessary to achieve a particular information processing goal: no constraints are imposed by this Reference Model either on the form of these interactions or on the possible relationships that may exist between them.

4.2.6 The activity of a given application process is represented by one or more application process invocations. Cooperation between application processes takes place via relationships established among application process invocations. At a particular time, an application process may be represented by none, one or more application process invocations. An application process invocation is responsible for coordinating its interactions with other application process invocations. Such coordination is outside the scope of this Reference Model.

4.2.7 OSI is concerned with the exchange of information between open systems (and not the internal functioning of each individual real open system).

4.2.8 As shown in Figure 1, the physical media for Open Systems Interconnection provides the means for the transfer of information between open systems.

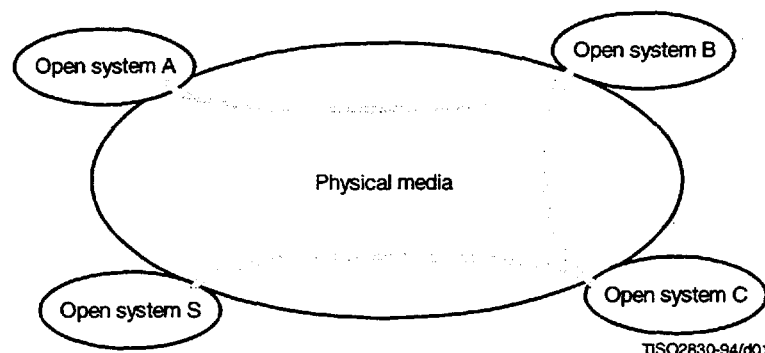


Figure 1 – Open systems connected by physical media

4.2.9 OSI is concerned only with the interconnection of systems. All other aspects of systems which are not related to interconnection are outside the scope of OSI.

4.2.10 OSI is concerned not only with the transfer of information between systems, i.e. transmission, but also with their capability to interwork to achieve a common (distributed) task. In other words, OSI is concerned with the interconnection aspects of cooperation¹⁾ between systems, which is implied by the expression "systems interconnection."

4.2.11 The objective of OSI is to define a set of standards to enable real open systems to cooperate. A system which complies with the requirements of applicable OSI standards in its cooperation with other systems is termed a real open system.

4.2.12 The design intent of the OSI standards is to specify a set of standards that make it possible for autonomous systems to communicate. Any equipment which communicates in conformance with all applicable OSI protocol standards is a real world equivalent of the model concept "open system". Equipment that is in the "terminal" category, that is, one that requires human intervention for the dominant parts of information processing, may satisfy the conditions of the previous sentences when the appropriate OSI standards are employed in communication with other open systems.

4.3 Modelling the OSI Environment

4.3.1 The development of OSI standards, i.e. standards for the interconnection of real open systems, is assisted by the use of abstract models. To specify the external behavior of interconnected real open systems, each real open system is replaced by a functionally equivalent abstract model of a real open system called an open system. Only the interconnection aspects of these open systems would strictly need to be described. However to accomplish this, it is necessary to describe both the internal and external behavior of these open systems. Only the external behavior of open systems is retained for the definition of standards for real open systems. The description of the internal behavior of open systems is provided in the Basic Reference Model only to support the definition of the interconnection aspects. Any real system which behaves externally as an open system can be considered to be a real open system.

4.3.2 This abstract modelling is used in two steps.

4.3.3 First, basic elements of open systems and some key decisions concerning their organization and functioning, are developed. This constitutes the Basic Reference Model of Open Systems Interconnection described in this Recommendation | Part of this International Standard.

4.3.4 Then, the detailed and precise description of the functioning of the open system is developed in the framework formed by the Basic Reference Model. This constitutes the services and protocols for OSI which are the subject of other Recommendations and/or International Standards.

4.3.5 It should be emphasized that the Basic Reference Model does not, by itself, specify the detailed and precise functioning of the open system and, therefore, it does not specify the external behavior of real open systems and does not imply the structure of the implementation of a real open system.

4.3.6 The reader not familiar with the technique of abstract modelling is cautioned that those concepts introduced in the description of open systems constitute an abstraction despite a similar appearance to concepts commonly found in real systems. Therefore, real open systems need not be implemented as described by the Model.

¹⁾ Cooperation among open systems involves a broad range of activities of which the following have been identified:

- a) interprocess communication, which concerns the exchange of information and the synchronization of activity between OSI application processes;
- b) data representation, which concerns all aspects of the creation and maintenance of data descriptions and data transformations for reformatting data exchanged between open systems;
- c) data storage, which concerns storage media, and file and database systems for managing and providing access to data stored on the media;
- d) process and resource management, which concerns the means by which OSI application processes are declared, initiated and controlled, and the means by which they acquire OSI resources;
- e) integrity and security, which concern information processing constraints that have to be preserved or assured during the operation of the open systems; and
- f) program support, which concerns the definition, compilation, linking, testing, storage, transfer, and access to the programs executed by OSI application-processes.

Some of these activities may imply exchange of information between the interconnected open systems and their interconnection aspects may, therefore, be of concern to OSI.

This Basic Reference Model covers the elements of OSI aspects of these activities which are essential for early development of OSI standards.

4.3.7 Throughout the remainder of this Basic Reference Model, only the aspects of real systems and application-processes which lie within the OSI Environment (OSIE) are considered. Their interconnection is illustrated throughout this Reference Model as depicted in Figure 2.

4.3.8 The extent of the application of the OSIE concept through the use of OSI standards may result in subsets of the OSIE which corresponds to partially disjoint sets of real open systems, which are not physically capable of OSI communication between them.

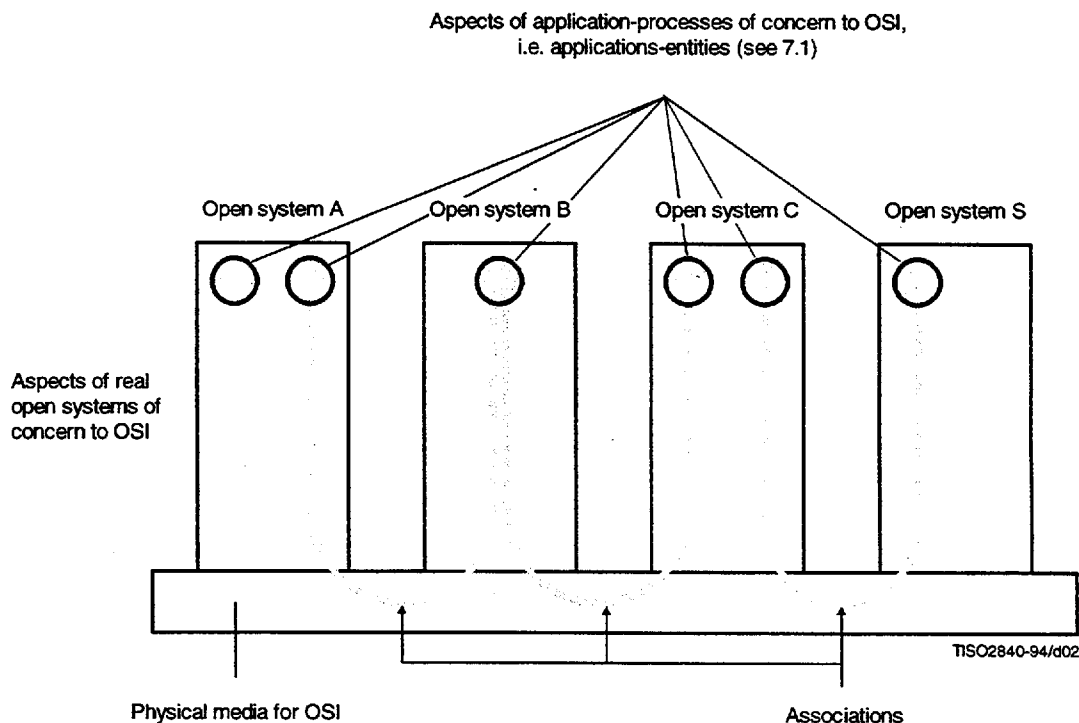


Figure 2 – Basic elements of OSI

5 Concepts of a layered architecture

5.1 Introduction

5.1.1 Clause 5 sets forth the architectural concepts that are applied in the development of the Reference Model of Open Systems Interconnection. Firstly, the concept of a layered architecture (with layers, entities, service-access-points, protocols, connections, etc.) is described. Secondly, identifiers are introduced for entities, service-access-points, and connections. Thirdly, service-access-points and data-units are described. Fourthly, elements of layer operation are described including connections, transmission of data, and error functions. Then, routing aspects are introduced and finally, management aspects are discussed.

5.1.2 The concepts described in clause 5 are those required to describe the Reference Model of Open Systems Interconnection. However, not all of the concepts described are employed in each layer of the Reference Model.

5.1.3 Four elements are basic to the Reference Model (see Figure 2):

- a) open systems;
- b) the application-entities which exist within the OSI Environment (see 7.1);
- c) the associations (see 5.3) which join the application-entities and permit them to exchange information; and
- d) the physical media for OSI.

NOTE – Security aspects which are also general architectural elements of protocols are discussed in CCITT Rec. X.800 I ISO 7498-2.

5.2 Principles of layering

5.2.1 Definitions

5.2.1.1 (N)-subsystem: An element in a hierarchical division of an open system which interacts directly only with elements in the next higher division or the next lower division of that open system.

5.2.1.2 (N)-layer: A subdivision of the OSI architecture, constituted by subsystems of the same rank (N).

5.2.1.3 peer-(N)-entities: Entities within the same (N)-layer.

5.2.1.4 sublayer: A subdivision of a layer.

5.2.1.5 (N)-service: A capability of the (N)-layer and the layers beneath it, which is provided to (N+1)-entities at the boundary between the (N)-layer and the (N+1)-layer.

5.2.1.6 (N)-facility: A part of an (N)-service.

5.2.1.7 (N)-function: A part of the activity of (N)-entities.

5.2.1.8 (N)-service-access-point, (N)-SAP: The point at which (N)-services are provided by an (N)-entity to an (N+1)-entity.

5.2.1.9 (N)-protocol: A set of rules and formats (semantic and syntactic) which determines the communication behavior of (N)-entities in the performance of (N)-functions.

5.2.1.10 (N)-entity-type: A description of a class of (N)-entities in terms of a set of capabilities defined for the (N)-layer.

5.2.1.11 (N)-entity: An active element within an (N)-subsystem embodying a set of capabilities defined for the (N)-layer that corresponds to a specific (N)-entity-type (without any extra capabilities being used).

5.2.1.12 (N)-entity-invocation: A specific utilization of part or all of the capabilities of a given (N)-entity (without any extra capabilities being used).

5.2.2 Description

5.2.2.1 The basic structuring technique in the Reference Model of Open Systems Interconnection is layering. According to this technique, each open system is viewed as logically composed of an ordered set of (N)-subsystems, represented for convenience in the vertical sequence shown in Figure 3. Adjacent (N)-subsystems communicate through their common boundary. (N)-subsystems of the same rank (N) collectively form the (N)-layer of the Reference Model of Open Systems Interconnection. There is one and only one (N)-subsystem in an open system for layer N. An (N)-subsystem consists of one or several (N)-entities. Entities exist in each (N)-layer. Entities in the same (N)-layer are termed peer-(N)-entities. Note that the highest layer does not have an (N+1)-layer above it and the lowest layer does not have an (N-1)-layer below it.

5.2.2.2 Not all peer-(N)-entities need or even can communicate. There may be conditions which prevent this communication (for example: they are not in interconnected open systems, or they do not support the same protocol subsets). Communication among peer-(N)-entities which reside in the same (N)-subsystem is provided by the LSE and therefore is out of the scope of OSI.

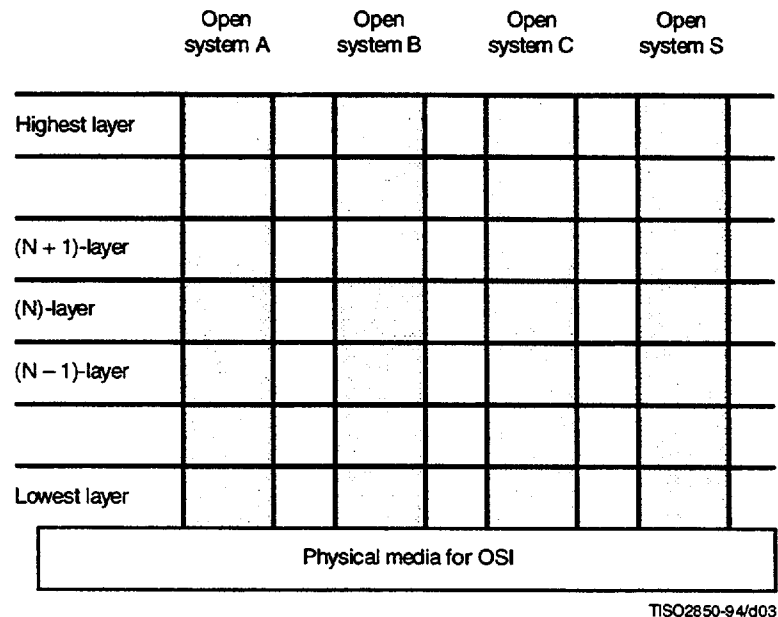


Figure 3 – Layering in cooperating open systems

NOTES

1 The distinction between the type of some object and an instance of that object is a distinction of significance for OSI. A type is a description of a class of objects. An instance of this type is any object that conforms to this description. The instances of the same type constitute a class. A type, and any instances of this type can be referred to by an individual name. Each nameable instance and the type to which this instance belongs carry distinguishable names.

For example, given that a programmer has written a computer program, that programmer has generated a type of something where instances of that type are created every time that particular program is invoked into execution by a computer. Thus, a FORTRAN compiler is a type and each occasion where a copy of that program is invoked in a data processing machine one displays an instance of that program.

The general concept of instantiation applies within OSI: Consider now an (N)-entity in the OSI context. It too, has two aspects, a type and a collection of invocations. The type of an (N)-entity is defined by description of the specific set of (N)-layer functions it is able to perform. An invocation of that type of (N)-entity is a specific invocation of whatever it is within the relevant open system that provides the (N)-layer functions called for by its type for a particular occasion of communication. It follows from these observations that (N)-entities refer only to the properties of an association between peer (N)-entities, while an (N)-entity-invocation refers to the specific, dynamic occasions of actual information exchange.

It is important to note that actual communication occurs only between (N)-entity-invocations at all layers. In the connection-mode (see 5.3.3), it is only at connection establishment time (or its logical equivalent during a recovery process) that (N)-entities are explicitly relevant. An actual connection is always made with a specific (N)-entity-invocation, although the request for connection is often made to an arbitrary (N)-entity (of a specific type). If an (N)-entity-invocation is aware of the name of its peer-(N)-entity-invocation, it is able to request another connection to that (N)-entity-invocation.

2 It may be necessary to further divide a layer into small substructures called sublayers and to extend the technique of layering to cover other dimensions of OSI. A sublayer is defined as a grouping of functions in a layer which may be bypassed. The bypassing of all the sublayers of a layer is not allowed. A sublayer uses the entities and communication services of the layer. The detailed definition or additional characteristics of a sublayer are for further study.

5.2.2.3 Except for the highest layer, each (N)-layer provides (N+1)-entities in the (N+1)-layer with an (N)-service at (N)-SAP(s). The properties of (N)-SAPs are described in 5.5. The highest layer is assumed to represent all possible uses of the (N)-service which are provided by the lower layers.

NOTE – Not all open systems provide the initial source or final destination of data. Such open systems need not contain the higher layers of the architecture (see Figure 12).

5.2.2.4 Each service provided by an (N)-layer may be tailored by the selection of one or more (N)-facilities which determine the attributes of that service. When a single (N)-entity cannot by itself fully support a service requested by an (N+1)-entity it calls upon the co-operation of other (N)-entities to help complete the service request. In order to co-operate, (N)-entities in any layer, other than those in the lowest layer, communicate by means of the set of services provided by the (N-1)-layer (see Figure 4). The entities in the lowest layer are assumed to communicate directly via the physical media which connect them.

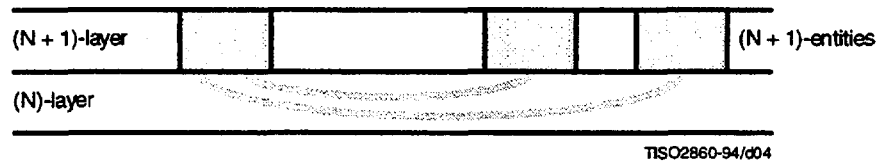


Figure 4 – (N + 1)-entities in the (N + 1)-layer communicate through the (N)-layer

5.2.2.5 The services of an (N)-layer are provided to the (N+1)-layer using the (N)-functions performed within the (N)-layer and as necessary the services available from the (N-1)-layer.

NOTE – This does not preclude the case where no protocol action is required in the (N)-layer to support a given (N)-facility because it is already available at the (N-1)-service boundary. However, null functionality of the complete (N)-protocol is not allowed.

5.2.2.6 An (N)-entity may provide services to one or more (N+1)-entities and use the services of one or more (N-1)-entities. An (N)-service-access-point is the point at which a pair of entities in adjacent layers use or provide services (see Figure 7).

5.2.2.7 Cooperation between (N)-entities is governed by one or more (N)-protocols. The entities and protocols within a layer are illustrated in Figure 5.

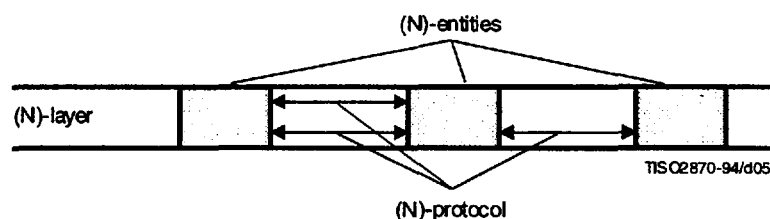


Figure 5 – (N)-protocols between (N)-entities

5.3 Communication between peer-entities

5.3.1 Definitions

- 5.3.1.1 (N)-association:** A cooperative relationship among (N)-entity-invocations.
- 5.3.1.2 (N)-connection:** An association requested by an (N+1)-entity for the transfer of data between two or more (N+1)-entities. The association is established by the (N)-layer and provides explicit identification of a set of (N)-data-transmissions and agreement concerning the (N)-data-transmission services to be provided for the set.
- 5.3.1.3 (N)-connection-endpoint:** A terminator at one end of an (N)-connection within an (N)-service-access-point.
- 5.3.1.4 multi-endpoint-connection:** A connection with more than two connection-endpoints.
- 5.3.1.5 correspondent (N)-entities:** (N)-entities with an (N-1)-connection between them.
- 5.3.1.6 (N)-relay:** An (N)-function by means of which an (N)-entity forwards data received from one peer-(N)-entity to another peer-(N)-entity.
- 5.3.1.7 (N)-data-source:** An (N)-entity that sends (N-1)-service-data-units (see 5.6.1.7) on an (N-1)-connection.²⁾
- 5.3.1.8 (N)-data-sink:** An (N)-entity that receives (N-1)-service-data-units on an (N-1)-connection²⁾
- 5.3.1.9 (N)-data-transmission:** An (N)-facility which conveys (N)-service-data-units from one (N+1)-entity to one or more (N+1)-entities.
- 5.3.1.10 (N)-duplex-transmission:** (N)-data-transmission in both directions at the same time.²⁾
- 5.3.1.11 (N)-half-duplex-transmission:** (N)-data-transmission in either direction, one direction at a time; the choice of direction is controlled by an (N+1)-entity.²⁾
- 5.3.1.12 (N)-simplex-transmission:** (N)-data-transmission in one pre-assigned direction.²⁾
- 5.3.1.13 (N)-data-communication:** An (N)-function which transfers (N)-protocol-data-units (see 5.6.1.3) according to an (N)-protocol, over one or more (N-1)-connections.²⁾
- 5.3.1.14 (N)-two-way-simultaneous-communication:** (N)-data-communication in both directions at the same time.
- 5.3.1.15 (N)-two-way-alternate-communication:** (N)-data-communication in both directions, one direction at a time.
- 5.3.1.16 (N)-one-way-communication:** (N)-data-communication in one pre-assigned direction.
- 5.3.1.17 (N)-connection-mode transmission:** (N)-data-transmission in the context of an (N)-connection.
- 5.3.1.18 (N)-connectionless-mode transmission:** (N)-data-transmission not in the context of an (N)-connection and not required to maintain any logical relationship between (N)-service-data-units.

5.3.2 Description

5.3.2.1 For information to be exchanged between two or more (N+1)-entities, an association is established between them in the (N)-layer using an (N)-protocol.

NOTE – Classes of protocols may be defined within the (N)-protocols.

5.3.2.2 The rules and formats of an (N)-protocol are instantiated in an (N)-subsystem by an (N)-entity. An (N)-entity may support one or more (N)-protocols. (N)-entities may support (N)-protocols which are connection-mode or connectionless-mode or both. (N)-entities when supporting connection-mode maintain the binding of (N)-connections to the appropriate (N+1)-entities at the appropriate (N)-SAPs. (N)-entities when supporting connectionless-mode maintain a binding with the appropriate (N)-SAPs for delivering the connectionless data to the (N+1)-entities.

5.3.2.3 (N+1)-entities can communicate only by using the services of the (N)-layer. There are instances where services provided by the (N)-layer do not permit direct access between all of the (N+1)-entities which have to communicate. If this is the case communication can still occur if some other (N+1)-entities can act as relays between them (see Figure 6).

²⁾ These definitions are not for use in this Basic Reference Model, but are for use in other OSI standards.

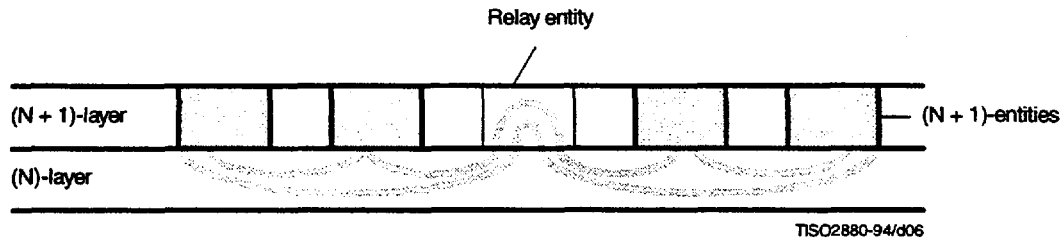


Figure 6 – Communication through a relay

5.3.2.4 The fact that communication is relayed by a chain of (N+1)-entities is known neither by the (N)-layer nor by the (N+2)-layer.

5.3.3 Modes of communication

5.3.3.1 Introduction

5.3.3.1.1 An (N)-layer may offer a connection-mode service, a connectionless-mode service, or both, to the (N+1)-layer, using the service or services provided by the (N-1)-layer. Any instance of transmission between the (N+1)-entities must utilize the same mode of (N)-service.

5.3.3.1.2 Both the (N)-connection-mode service and the (N)-connectionless-mode service are characterised by the facilities which they offer to, and the quality of service seen by, the (N+1)-entities. For both the (N)-connection-mode service and the (N)-connectionless-mode service, functions may be provided by the (N)-layer to enhance the facilities offered to, and the quality of service seen by the (N+1)-entities over those which are offered to the (N)-layer by the (N-1)-layer and, if necessary, to convert between one mode of service and another.

5.3.3.1.3 Since connection-mode transmission and connectionless-mode transmission are complementary concepts, they are best understood in juxtaposition, particularly since connectionless-mode transmission is defined most easily in relationship to the concept of a connection.

5.3.3.1.4 In order for (N+1)-entities to be able to communicate using an (N)-connection-mode service or an (N)-connectionless-mode service it is essential that a pre-arranged association exists between them, constituted by the pre-knowledge which it is essential that each (N+1)-entity has of the others in order at least to initiate the use of the service. This association is established in ways which are not detailed in this Basic Reference Model and comprises four elements:

- a) knowledge of the addresses of the peer-(N)-entities involved;
- b) knowledge of a protocol agreed by the peer-(N)-entities for use at least to initiate communication;
- c) knowledge of the availability for communication of the peer-(N)-entities;
- d) knowledge of the quality of service available from the (N)-service.

NOTE – The pre-knowledge constituting a pre-arranged association can be acquired in many ways; some examples are listed below:

- a) from information acquired manually when contracts are exchanged with a service provider;
- b) from information which a network administration may provide in a directory or enquiry database;
- c) from information that may be learned from previous instances of communication;
- d) from information that may be provided dynamically through the operation of management protocols.

The total pre-knowledge constituting a pre-arranged association is likely to be acquired in a combination of the above ways.

5.3.3.2 Connection mode

5.3.3.2.1 A connection is an association established for the transfer of data between two or more peer-(N)-entities. This association binds the peer-(N)-entities together with the (N-1)-entities in the next lower layer. The ability to establish and release a connection and to transfer data over it is provided to the (N)-entities in a given (N)-layer by the next lower layer as a connection-mode service. The use of a connection-mode service by peer-(N)-entities proceeds through three distinct phases:

- a) connection establishment;
- b) data transfer; and
- c) connection release.

5.3.3.2.2 In addition to the clearly distinguishable lifetime exhibited by these phases, a connection has the following fundamental characteristics:

- a) it involves establishing and maintaining a two or more party agreement concerning the transmission of data among the peer-(N)-entities concerned, and using the provider of the (N-1)-service;
- b) it allows the negotiation between all the parties concerned of the parameters and options that will govern the transmission of data;
- c) it provides connection identification by means of which the overheads involved in address resolution and transmission can be avoided on data transfers;
- d) it provides a context within which successive units of data transmitted between the peer-entities are logically related, and makes it possible to maintain sequence and provide flow control for those transmissions.

5.3.3.2.3 The characteristics of connection-mode transmission are particularly attractive in applications which call for relatively long-lived, stream-oriented interactions between entities in stable configurations. Examples are provided by direct terminal use of a remote computer, file transfer, and long-term attachment of remote job entry stations. In these cases, the entities involved initially discuss their requirements and agree to the terms of their interaction, reserving whatever resources they may need, transfer a series of related units of data to accomplish their mutual objective, and explicitly end their interaction, releasing the previously reserved resources. The properties of connection-mode transmission are also relevant in a wide range of other applications.

5.3.3.2.4 Connection-mode transmission is accomplished through the use of (N)-connections. (N)-connections are provided by the (N)-layer between two or more (N)-service-access-points. The terminator of an (N)-connection at an (N)-service-access-point is called an (N)-connection-endpoint. An (N)-connection is provided by the (N)-layer between two or more (N)-service-access-points at the request of a calling (N+1)-entity in support of the (N+1)-entities attached to the (N)-service-access-points involved in the (N)-connection. An (N)-connection with more than two endpoints is termed a multi-endpoint-connection. (N)-entities with a connection between them are termed correspondent (N)-entities.

NOTE – Data transfer using an (N)-connection-mode service involves the establishment of an (N)-connection prior to the data transfer. This dynamically sets up an association between the (N+1)-entities and the (N)-connection-mode service in addition to the association identified in 5.3.2. This association involves elements which are not part of the pre-arranged association described in 5.3.3.1.4, in particular:

- a) knowledge of the willingness of the peer-(N)-entity or entities to undertake a specific communication, and of the willingness of the underlying service to support it and;
- b) the ability for the peer-(N)-entities to negotiate and renegotiate the characteristics of the communication.

5.3.3.3 Connectionless mode

5.3.3.3.1 Connectionless-mode transmission is the transmission of a single unit of data from a source service-access-point to one or more destination service-access-points without establishing a connection. A connectionless-mode service allows an entity to initiate such a transmission by the performance of a single service access.

5.3.3.3.2 In contrast to a connection, an instance of the use of a connectionless-mode service does not have a clearly distinguishable lifetime. In addition, it has the following fundamental characteristics:

- a) it requires only a pre-arranged association between the peer-(N)-entities involved which determines the characteristics of the data to be transmitted, and no dynamic agreement is involved in an instance of the use of the service;
- b) all the information required to deliver a unit of data – destination address, quality of service selection, options, etc. – is presented to the layer providing the connectionless-mode service, together with the unit of data to be transmitted, in a single service access. The layer providing the connectionless-mode service is not required to relate this access to any other access.

5.3.3.3.3 As a result of these fundamental characteristics it may also be true that:

- a) each unit of data transmitted is routed independently by the layer providing the connectionless-mode service; and
- b) copies of a unit of data can be transmitted to a number of destination addresses.

5.3.3.3.4 These characteristics of connectionless-mode transmission do not preclude making available to the service user information on the nature and quality of service which may apply for a single invocation of the service or which may be observed over successive invocations of the service between pairs of (N)-service-access-points or among a set of (N)-service-access-points.

5.3.3.3.5 For each layer, the subclauses of clause 7 identify those items which have relevance to the connectionless-mode service provided by that layer.

5.3.3.3.6 The basic (N)-connectionless-mode service is a service which meets the following conditions:

- a) it is not required to exhibit any minimum values of the quality of service measures, in particular the sequence of (N)-service-data-units need not be maintained and;
- b) it is not required to exhibit peer flow control.

5.3.3.3.7 Any (N)-connectionless-mode service definition should allow the basic service.

5.3.3.3.8 Since the basic service is not required to maintain the sequence of (N)-service-data-units, there is no requirement for any (N)-layer to provide sequencing functions. However, in real implementations the characteristics of the underlying medium or of real subnetworks may offer a high probability of in-sequence delivery and this may be reflected in the characteristics of the connectionless-mode services offered by higher layers.

5.3.3.3.9 An (N+1)-entity provides no information to the provider of an (N)-connectionless-mode service about the logical relationships between (N)-service-data-units, apart from the source and destination (N)-service-access-point-addresses.

5.3.3.3.10 From the point of view of the (N+1)-entity this means that it is not able to require the (N)-service to apply a particular function to a sequence of (N)-service-data-units sent by it. However, from the point of view of the (N)-layer, this does not imply any constraint on the functions which support the service.

5.3.3.3.11 (N+1)-entities can communicate using an (N)-connectionless-mode service provided that there is a pre-arranged association between them providing knowledge about each other which allows them to do so. This knowledge should allow the locations of the (N+1)-entities to be determined, it should determine the correct interpretation of (N)-service-data-units by a receiving (N+1)-entity, and it may define the rates of transfer, rates of response, and the protocol in use between the entities. The knowledge may result from prior agreement between the (N+1)-entities concerning the parameters, formats, and options to be used.

5.3.3.3.12 (N+1)-entities may require prior knowledge of the facilities offered by the service and the quality of service which they can expect to receive from it in order to choose an (N+1)-protocol to be used for communication over an (N)-connectionless-mode service.

5.3.4 The relationship between services provided at adjacent layer boundaries

5.3.4.1 There are no architectural constraints on any vertical combination of an (N)-layer providing one type of (N)-service (connection-mode or connectionless-mode) using the other type of (N-1)-service. In principle the services at the two layer boundaries can be:

- a) both connection-mode services;
- b) both connectionless-mode services;
- c) the (N)-service a connection-mode service and the (N-1)-service a connectionless-mode service;
- d) the (N)-service a connectionless-mode service and the (N-1)-service a connection-mode service.

5.3.4.2 In order to allow combinations c) and d) two architectural elements are required:

- a) a function to provide an (N)-connection-mode service using an (N-1)-connectionless-mode service; and
- b) a function to provide an (N)-connectionless-mode service using an (N-1)-connection-mode service.

These are known as mode conversion functions.

NOTE – Of these functions, function a) requires significant protocol-control-information. For example, there is a need to identify the connection which is constructed, control its state and provide sequencing of service-data-units. Function b) requires little or no additional protocol-control-information, rather, it places constraints on the way in which the connection-mode service is used.

5.3.5 Application of mode conversion functions

5.3.5.1 Mode conversion functions may be invoked in OSI end systems, or in OSI relay systems (see 6.5). When invoked in OSI relay systems, the mode conversion functions may either:

- a) join an (N)-protocol using the (N-1)-connectionless service and an (N)-protocol using the (N-1)-connection-mode service in support of an (N)-connection-mode service; or
- b) join an (N)-protocol using the (N-1)-connectionless-mode service and an (N)-protocol using the (N-1)-connection-mode service in support of an (N)-connectionless mode service.

5.3.5.2 The use of mode conversions between (N-1)-services within a layer is not explicitly constrained by the Reference Model but, where several (N-1)-services are connected in tandem, the use of mode conversions would be ordered to minimize the number of mode conversions necessary to arrive at a given composite (N)-service.

5.3.5.3 Where an (N-1)-connectionless-mode service is enhanced to provide a (N)-connection-mode service, a number of (N)-connections may be supported by (N-1)-connectionless-mode transmission between the same (N-1)-service-access-points.

5.3.5.4 Where an (N-1)-connection-mode service is used to provide an (N)-connectionless-mode service, (N)-connectionless-mode transmission between a number of different (N)-service-access-points may be supported by the same (N-1)-connection.

5.4 Identifiers

5.4.1 Definitions

5.4.1.1 (N)-address: A name unambiguous within the OSIE which is used to identify a set of (N)-service-access-points which are all located at a boundary between an (N)-subsystem and an (N+1)-subsystem in the same open system.

NOTE – A name is unambiguous within a given scope when it identifies one and only one object within that scope. Unambiguity of a name does not preclude the existence of synonyms.

5.4.1.2 (N)-service-access-point-address; (N)-SAP-address: An (N)-address that is used to identify a single (N)-SAP.

5.4.1.3 (N)-address-mapping: An (N)-function which provides the mapping between the (N)-addresses and the (N-1)-addresses associated with an (N)-entity.

5.4.1.4 routing: A function within a layer which translates the title of an entity or the service-access-point-address to which the entity is attached into a path by which the entity can be reached.

5.4.1.5 (N)-connection-endpoint-identifier: An identifier of an (N)-connection-endpoint which can be used to identify the corresponding (N)-connection at an (N)-service-access-point.

5.4.1.6 (N)-connection-endpoint-suffix: A part of an (N)-connection-endpoint-identifier which is unique within the scope of an (N)-service-access-point.

5.4.1.7 multi-connection-endpoint-identifier: An identifier which specifies the connection-endpoint of a multi-endpoint-connection which should accept the data that is being transferred.

5.4.1.8 (N)-service-connection-identifier: An identifier which uniquely specifies an (N)-connection within the environment of the correspondent (N+1)-entities.

5.4.1.9 (N)-protocol-connection-identifier: An identifier which uniquely specifies an individual (N)-connection within the environment of the multiplexed (N-1)-connection.

5.4.1.10 (N)-entity-title: A name that is used to identify unambiguously an (N)-entity.

5.4.2 Description

5.4.2.1 An (N)-service-access-point-address identifies a particular (N)-service-access-point to which an (N+1)-entity is attached (see Figure 7). When the (N+1)-entity is detached from the (N)-service-access-point, the (N)-SAP-address no longer provides access to the (N+1)-entity. If the (N)-service-access-point is reattached to a different (N+1)-entity, then the (N)-SAP-address identifies the new (N+1)-entity and not the old one.

5.4.2.2 The use of an (N)-SAP-address to identify an (N+1)-entity is the most efficient mechanism if the permanence of attachment between the (N+1)-entity and the (N)-service-access-point can be assured.

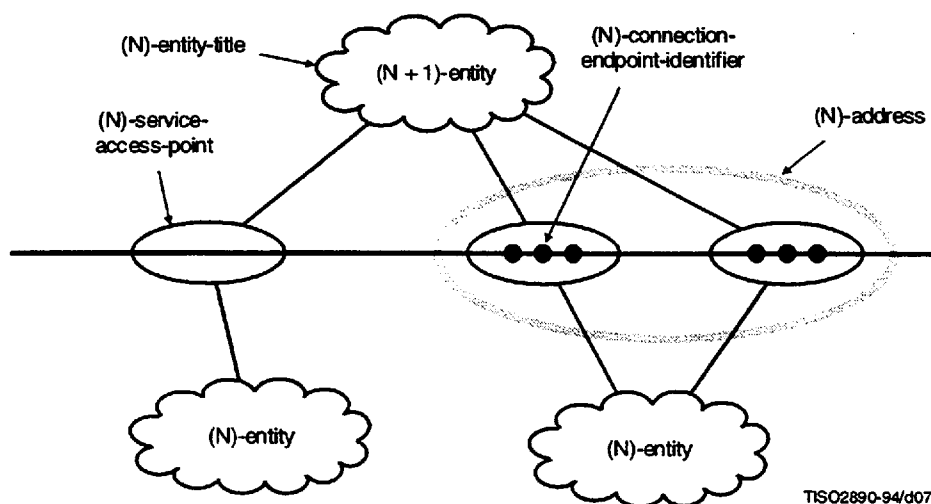


Figure 7 – Entities, service-access-points and identifiers

5.4.2.3 Interpretation of the correspondence between the (N)-addresses served by an (N)-entity and the (N-1)-addresses used for accessing (N-1)-services is performed by an (N)-address-mapping function.

5.4.2.4 The structure of an (N)-address is known by the (N)-entity which is attached to the identified (N)-service-access-point. However, the (N+1)-entity does not know this structure.

5.4.2.5 If an (N+1)-entity has two or more (N)-service-access-points with either the same (N)-entity or different (N)-entities, the (N)-entities have no knowledge of this fact. Each (N)-service-access-point is considered to identify a different (N+1)-entity from the perspective of the (N)-entities.

5.4.2.6 A routing function translates the (N)-address of an (N+1)-entity into a path or route by which the (N+1)-entity may be reached.

5.4.2.7 An (N+1)-entity may establish an (N)-connection with another (N+1)-entity by using an (N)-service. When an (N+1)-entity establishes an (N)-connection with another (N+1)-entity, each (N+1)-entity is given an (N)-connection-endpoint-identifier by its supporting (N)-entity. The (N+1)-entity can then distinguish the new connection from all other (N)-connections accessible at the (N)-service-access-point it is using. This (N)-connection-endpoint-identifier is unique within the scope of the (N+1)-entity which will use the (N)-connection.

5.4.2.8 The (N)-connection-endpoint-identifier consists of two parts:

- a) the (N)-SAP-address of the (N)-service-access-point which will be used in conjunction with the (N)-connection; and
- b) an (N)-connection-endpoint-suffix which is unique within the scope of the (N)-service-access-point.

5.4.2.9 A multi-endpoint-connection requires multi-connection-endpoint-identifiers. Each such identifier is used to specify which connection-endpoint should accept the data which is being transferred. A multi-connection-endpoint-identifier is unique within the scope of the connection within which it is used.

5.4.2.10 The (N)-layer may provide to the (N+1)-entities an (N)-service-connection-identifier which uniquely specifies the (N)-connection within the environment of the correspondent (N+1)-entities.

5.5 Properties of service-access-points

5.5.1 An (N+1)-entity requests (N)-service via an (N)-service-access-point which permits the (N+1)-entity to interact with an (N)-entity.

- 5.5.2** Both the (N)- and (N+1)-entities attached to an (N)-service-access-point are in the same system.
- 5.5.3** An (N+1)-entity may concurrently be attached to one or more (N)-service-access-points attached to the same or different (N)-entities.
- 5.5.4** An (N)-entity may concurrently be attached to one or more (N+1)-entities through (N)-service-access-points.
- 5.5.5** An (N)-service-access-point is attached to only one (N)-entity and to only one (N+1)-entity at a time.
- 5.5.6** An (N)-service-access-point may be detached from an (N+1)-entity and reattached to the same or another (N+1)-entity.
- 5.5.7** An (N)-service-access-point may be detached from an (N)-entity and reattached to the same or another (N)-entity.
- 5.5.8** An (N)-service-access-point is located by means of its (N)-SAP-address. An (N)-SAP-address is used by an (N+1)-entity to request an instance of communication.
- 5.5.9** An (N)-service-access-point may support:
- (N)-connection-mode service only;
 - (N)-connectionless-mode service only;
 - (N)-connection-mode services and (N)-connectionless-mode services concurrently.
- 5.5.10** A single (N+1)-entity may concurrently be using several (N)-connections and an (N)-connectionless-mode service through one or more (N)-service-access-points to which it is attached.
- 5.5.11** (N+1)-entities distinguish between instances of the (N)-connectionless-mode services and the (N)-connection-mode services offered concurrently through the same (N)-service-access-point by the uniqueness of the interactions prescribed for these services.

5.6 Data-units

5.6.1 Definitions

- 5.6.1.1 (N)-protocol-control information:** Information exchanged between (N)-entities to co-ordinate their joint operation.
- 5.6.1.2 (N)-user-data:** The data transferred between (N)-entities on behalf of the (N+1)-entities for whom the (N)-entities are providing services.
- 5.6.1.3 (N)-protocol-data-unit:** A unit of data specified in an (N)-protocol and consisting of (N)-protocol-control-information and possibly (N)-user-data.
- 5.6.1.4 (N)-service-data-unit:** An amount of information whose identity is preserved when transferred between peer-(N+1)-entities and which is not interpreted by the supporting (N)-entities.
- 5.6.1.5 expedited (N)-service-data-unit. (N)-expedited-data-unit:** A small (N)-service-data-unit whose transfer is expedited. The (N)-layer ensures that an expedited-data-unit will not be delivered after any subsequent service-data-unit or expedited unit sent on that connection.

5.6.2 Description

- 5.6.2.1** Information is transferred in various types of data-units between peer-(N)-entities. The data-units are defined in 5.6 and the relationships among them are illustrated in Figures 8 and 9.

	Control	Data	Combined
(N)-(N)-peer-entities	(N)-protocol-control-information	(N)-user-data	(N)-protocol-data-unit

Figure 8 – Relationships among data-units