SECTION 2

SERVICE DEFINITIONS

Recommendation X.210

Fascicle VIII-4 – Rec. X.210

OPEN SYSTEMS INTERCONNECTION
LAYER SERVICE DEFINITION CONVENTIONS[1)]

(Malaga-Torremolinos, 1984; amended at Melbourne, 1988)

The CCITT,

considering

(a) that Recommendation X.200 provides a Reference Model which separates the functions of inter-process communications into seven layers;

(b) that the layer services developed for these seven layers of function will generally result in separate Recommendations;

(c) that there is a need for the resulting Recommendations to be a coherent set,

unanimously recommends

that the conventions and terminology defined in this Recommendation be used for Layer Service definitions of Open Systems Interconnection.

CONTENTS

1 Scope and field of application

2 References

3 Definitions

4 Model for layer services

5 Service primitives

6 Conventions for time-sequence diagrams

Appendix I - Conventions for naming service primitives

Appendix II - Differences between Recommendation X.210 and ISO Technical Report 8509

1 Scope and field of application

This Recommendation establishes definitions of terms and conventions for reference by other Recommendations which define the services provided by layers 1 to 6 of the Reference Model of Open Systems Interconnection for CCITT Applications (Recommendation X.200). These conventions are also applicable in specifying other telecommunications and data processing services and capabilities which utilize the layered model concepts and principles prescribed by the Reference Model. In particular it is concerned with conventions relating to just two communicating systems at any one time, and connection-oriented services.

[1)] Recommendation X.210 and ISO TR 8509, [Information processing systems - Open Systems Interconnection - Service conventions] were developed in close collaboration and are technically aligned, except for the differences noted in Appendix II.

2 References

Recommendation X.200 - Reference Model of Open Systems Interconnection for CCITT Applications (see also ISO 7498).

3 Definitions

3.1 This Recommendation builds on the concepts developed in Recommendation X.200 and makes use of the following terms defined in that Recommendation:

a) (N) layer;

b) (N) service;

c) (N) entity;

d) (N) service-access-point;

e) (N) service-access-point-address.

Note – The term "service-access-point" is used when describing the relationship between primitives associated with a single connection. Further study is required to include the concept of connection endpoints in this description.

3.2 For the purpose of this Recommendation, the following definitions also apply:

3.2.1 **service-user**

An abstract representation of the totality of those entities in a single system that make use of a service through a single access point.

3.2.2 **service-provider**

An abstract machine which models the behaviour of the totality of the entities providing the service, as viewed by the user.

3.2.3 **service-primitive; primitive**

An abstract, implementation independent interaction between a service-user and the service-provider.

3.2.4 **request (primitive)**

A primitive issued by a service-user to invoke some procedure.

3.2.5 **indication (primitive)**

A primitive issued by a service-provider either:

i) to invoke some procedure; or

ii) to indicate that a procedure has been invoked by the service-user at the peer service-access-point.

3.2.6 **response (primitive)**

A primitive issued by a service-user to complete, at a particular service-access-point, some procedure previously invoked by an indication at that service-access-point.

3.2.7 **confirm (primitive)**

A primitive issued by a service-provider to complete, at a particular service-access-point, some procedure previously invoked by a request at that service-access-point.

Note – Confirms and responses can be positive or negative as appropriate to the circumstances.

3.2.8 **(N)-mandatory-service**

A service which must be provided in the (N)-service.

3.2.9 **(N)-provider-optional-service**

A service which may or may not be provided in the (N)-service.

3.2.10 **(N)-user-optional-service**

A service which will only be provided if the (N)-service-user requests it, and it is available in the (N)-service.

3.2.11 **unconfirmed-service**

A service which does not result in an explicit confirmation.

3.2.12 **confirmed-service**

A service which results in an explicit confirmation from the service-provider. There is not necessarily any relationship to a response from the peer service-user.

3.2.13 **provider-intitiated-service**

A service which is generated by the service-provider.

4 Model for layer services

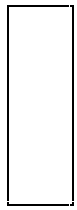
A layer service is defined in terms of an abstract model having the following elements:

a) (N)-service-users;

b) (N)-service-provider.

For the lifetime of a particular connection each service-user gains access to the service-provider as indicated in Figure 1/X.210.

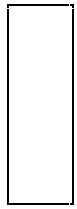
FIGURE 1/X.210



Each service-user interacts with the service-provider by issuing or receiving service-primitives. The layer service defines relations between interactions at one service-access-point and consequential interactions at service-access-points used by service-users in order to communicate.

The relationship among the terms service, boundary, service primitive, peer protocol, and peer entities are illustrated in Figure 2/X.210.

FIGURE 2/X.210



5 Service primitives

Note – The detailed properties of service primitives are for further study.

5.1 *General*

The use of primitives does not preclude any specific implementation of a service in terms of interface primitives. The following comments apply to this definition technique based on service primitives:

a) service primitives are conceptual, and need not be either directly related to protocol elements, or seen as macro calls of an access method to the layer service;

b) there are other equivalent sets of service primitives which can describe the same layer service;

c) only service primitives which correspond to some element of the layer service involving two service-users need to be considered. The primitives which are only related to local conventions between the service-user and service-provider do not relate to this description technique. For example, strictly local functions could be provided in some implementations. As they do not involve both users, such functions are not visible outside the local system.

5.2 *Categories of service*

The following types of service are identified:

a) mandatory-service (see § 3.2.8);

b) provider-optional-service (see § 3.2.9);

c) user-optional-service (see § 3.2.10).

A user optional service may be either a mandatory service or a provider optional service.

5.3 *Types of service primitives*

Four types of service primitives are identified:

a) request primitive (see § 3.2.4);

b) indication primitive (see § 3.2.5);

c) response primitive (see § 3.2.6);

d) confirm primitive (see § 3.2.7).

5.4 *Properties of primitives*

An individual service primitive is a logically separate interaction which cannot be interrupted by another interaction. A service primitive has a direction which is either:

a) from a service-user to the service-provider;

b) from the service-provider to a service-user.

One or more parameters may be associated with a service primitive and each of these parameters has a defined range of values. Parameter values associated with a service primitive are passed in the direction of the service primitive.

5.5 *Names of primitives*

The name of each service primitive contains three elements:

a) an initial (or initials) which specifies the layer (see § I.1);

b) a name which specifies the service-name (see § I.2);

c) a name which specifies the type of primitive (see § I.3).

6 Conventions for time-sequence diagrams

Time-sequence diagrams are used to illustrate how sequences of interactions are related in time.

Time-sequence diagrams (see Figure 3/X.210) indicate:

a) the sequence of events at each user/provider interface;

b) where appropriate, the sequence of events between peer users.

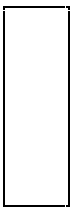
Each diagram is partitioned by two vertical lines into three fields. The central field represents the service-provider and the two side fields represent the two service-users. The lines represent the service-access-points between the service-users and the service-provider.

Sequences of events at each service-access-point are positioned along lines representing the passage of time, increasing downwards. Arrows, placed in the areas representing the service-user, indicate the direction of propagation of primitives (i.e., *to* or *from* the service-user) and may include implicit flow control between the service-user and service-provider.

Necessary sequence relations between the two interaction points are emphasized by a solid line between the time lines (e.g., in Figure 3a/X.210 the request primitive from one service-user to the service-provider at time t_1 is necessarily followed by the indication primitive to the peer service-user at time t_2). In the absence of this solid line, there is no specific relationship between the delivery of confirmation and indication. The absence of relationship is indicated either by leaving the central field blank or, for clarity, by use of a tilde (ñ).

Figures 3b and 3c/X.210 present alternative methods of indicating negative acknowledgements generated by the responding service-user. In Figure 3b/X.210 the same name (e.g., X) is used throughout the complete sequence, whereas in Figure 3c/X.210 the responding service-user employs a request with a different name (e.g., Y).

FIGURE 3/X.210



APPENDIX I

(to Recommendation X.210)

Conventions for naming service primitives

(This Appendix is not an integral part of the body of the Recommendation. It provides information for the authors of service Recommendations but is not necessary for users of service Recommendations.)

I.1 *Initials*

The following initials are used to specify Application services and the layers of the OSI Model:

a) P – Presentation Layer;

b) S – Session Layer;

c) T – Transport Layer;

d) N – Network Layer (see note 1);

e) DL – Data-Link Layer;

f) Ph – Physical Layer.

Note 1 – The use of ‘N' to signify the Network Layer is not to be confused with the use of ‘(N)-' to signify a particular but unspecified layer of the Model.

Note 2 – Selection of initials for the various types of Application services requires further study.

I.2 *Service name*

A single word consisting of the infinitive form of a verb is recommended for the service name (e.g., CONNECT, ABORT).

I.3 *Name of primitive type*

The name of the primitive type consists of one of the following (indicating the type of the primitive):

a) request;

b) indication;

c) response (positive or negative);

d) confirm (positive or negative).

I.4 *Representation*

The initial(s) is represented in the form given in § I.1. The service name is written in capital letters and the name of the primitive type is written in lower case letters.

The initial(s) and the service name are separated by a hyphen; the service name and primitive type are separated by a space.

I.5 *Examples*

The following are examples of primitive names which use these conventions:

a) P-CONNECT request;

b) T-DATA indication;

c) S-DISCONNECT confirm.

APPENDIX II

(to Recommendation X.210)

Differences between Recommendation X.210 and

ISO Technical Report 8509

II.1 CCITT Recommendation X.210 and ISO Technical Report 8509 are technically aligned to the extent that ISO layer service definitions using these conventions are not affected. However, there are a number of detailed wording differences that remain to be reconciled.