

INTERNATIONAL STANDARD

ISO/IEC
8326

Second edition
1996-09-15

Information technology — Open Systems Interconnection — Session service definition

*Technologies de l'information — Interconnexion de systèmes ouverts
(OSI) — Définition du service de session*



Reference number
ISO/IEC 8326:1996(E)

AVA Documents
Document Sharing Hub

Contents

	<i>Page</i>
SECTION 1 – GENERAL.....	1
1 Scope.....	1
2 Normative references	1
2.1 Identical Recommendations International Standards	1
2.2 Paired Recommendations International Standards equivalent in technical content	2
3 Definitions.....	2
3.1 Reference Model definitions	2
3.2 Service convention definitions.....	2
3.3 Session service definitions.....	2
4 Abbreviations	3
4.1 Data units	3
4.2 Miscellaneous	3
4.3 Service variables	3
5 Conventions.....	3
6 Model of the session service	4
6.1 Model of the layer service.....	4
6.2 Model of session connectionless-mode transmission.....	4
7 Overview of the session service	4
7.1 General overview	4
7.2 Token concept.....	4
7.3 Synchronization and dialogue unit concepts.....	5
7.4 Activity concept.....	5
7.5 Resynchronization.....	6
7.6 Negotiation.....	6
7.6.1 Negotiation of functional units	6
7.6.2 Negotiation of initial token settings	7
7.6.3 Negotiation of initial synchronization point serial number.....	7
8 Phases and services of the session service	7
8.1 Session connection establishment phase.....	7
8.2 Data transfer phase.....	7
8.3 Session connection release phase.....	9

© ISO/IEC 1996

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

9	Functional units and subsets.....	9
9.1	Functional units.....	9
9.1.1	Kernel functional unit	10
9.1.2	Negotiated release functional unit.....	10
9.1.3	Half-duplex functional unit.....	11
9.1.4	Duplex functional unit	11
9.1.5	Expedited data functional unit	11
9.1.6	Typed data functional unit	11
9.1.7	Capability data exchange functional unit.....	11
9.1.8	Minor synchronize functional unit.....	11
9.1.9	Symmetric synchronize functional unit	11
9.1.10	Data separation functional unit	11
9.1.11	Major synchronize functional unit	11
9.1.12	Resynchronize functional unit	11
9.1.13	Exceptions functional unit	11
9.1.14	Activity management functional unit.....	11
9.2	Subsets	12
10	Quality of session service.....	12
10.1	Determination of QOS	12
10.2	Session connection QOS negotiation procedures	13
10.3	Definition of QOS parameters	14
10.3.1	Session connection establishment delay	14
10.3.2	Session connection establishment failure probability	14
10.3.3	Throughput.....	15
10.3.4	Transit delay.....	15
10.3.5	Residual error rate.....	15
10.3.6	Transfer failure probability	16
10.3.7	Session connection release delay	16
10.3.8	Session connection release failure probability	17
10.3.9	Session connection protection.....	17
10.3.10	Priority	17
10.3.11	Session connection resilience.....	17
10.3.12	Extended control parameter	17
10.3.13	Optimized dialogue transfer.....	17
SECTION 2 – DEFINITION OF CONNECTION-ORIENTED SESSION SERVICE PRIMITIVES		18
11	Introduction to session service primitives	18
11.1	Summary of primitives.....	18
11.2	Token restrictions on sending primitives	20
11.3	Sequencing of primitives	21
11.4	Synchronization point serial number management	21
11.4.1	Management of single serial synchronization point numbers.....	22
11.4.2	Management of symmetric serial number synchronization point numbers	24
12	Session connection establishment phase	27
12.1	Session connection service.....	27
12.1.1	Function	27
12.1.2	Types of primitives and their parameters.....	27
12.1.3	Sequence of primitives.....	30
13	Data transfer phase	30
13.1	Normal data transfer service	30
13.1.1	Function	30

13.1.2	Types of primitives and their parameters.....	30
13.1.3	Sequence of primitives.....	30
13.2	Expedited data transfer service	31
13.2.1	Function	31
13.2.2	Types of primitives and their parameters.....	31
13.2.3	Sequence of primitives.....	31
13.3	Typed data transfer service	32
13.3.1	Function	32
13.3.2	Types of primitives and their parameters.....	32
13.3.3	Sequence of primitives.....	32
13.4	Capability data exchange service.....	32
13.4.1	Function	32
13.4.2	Types of primitives and their parameters.....	32
13.4.3	Sequence of primitives.....	33
13.5	Give tokens service.....	33
13.5.1	Function	33
13.5.2	Types of primitives and their parameters.....	33
13.5.3	Sequence of primitives.....	34
13.6	Please tokens service.....	34
13.6.1	Function	34
13.6.2	Types of primitives and their parameters.....	34
13.6.3	Sequence of primitives.....	35
13.7	Give control service	35
13.7.1	Function	35
13.7.2	Types of primitives and their parameters.....	35
13.7.3	Sequence of primitives.....	36
13.8	Minor synchronization point service.....	36
13.8.1	Function	36
13.8.2	Types of primitives and their parameters.....	37
13.8.3	Sequence of primitives.....	37
13.9	Major synchronization point service.....	38
13.9.1	Function	38
13.9.2	Types of primitives and their parameters.....	39
13.9.3	Sequence of primitives.....	39
13.10	Resynchronize service	40
13.10.1	Function	40
13.10.2	Types of primitives and their parameters.....	41
13.10.3	Sequence of primitives.....	42
13.11	P-exception reporting service.....	43
13.12	U-exception-reporting service.....	44
13.13	Activity start service	46
13.14	Activity resume service.....	47
13.15	Activity interrupt service	48
13.16	Activity discard service.....	49
13.17	Activity end service	51
14	Session connection release phase.....	52
14.1	Orderly release service.....	52
14.2	U-abort service.....	53
14.3	P-abort service	54
15	Sequences of primitives	55
15.1	State tables	55
15.2	Sequences of primitives at one session connection end-point	55

16	Collision	55
16.1	Collision as viewed by the SS-user	55
16.2	Collision resolution by the SS-provider	55
SECTION 3 – DEFINITION OF CONNECTIONLESS-MODE SESSION SERVICE PRIMITIVES		57
17	S-UNIT-DATA	57
17.1	Function	57
17.2	Types of Primitives and Parameters	57
17.3	Sequence of Primitives	58
Annex A – State tables		59
A.1	General	59
A.2	Notation for state tables	59
A.3	Conventions for entries in state tables	59
A.4	Actions to be taken by the SS-user	60
A.5	Definitions of sets and variables	60

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 8326 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 21, *Open Systems Interconnection, data management and open distributed processing*, in collaboration with ITU-T. The identical text is published as ITU-T Recommendation X.215.

This second edition cancels and replaces the first edition (ISO 8326:1987), which has been technically revised. It also incorporates Amendment 4:1992.

Annex A forms an integral part of this International Standard.

Introduction

This Recommendation | International Standard is one of a set of Recommendations | International Standards produced to facilitate the interconnection of computer systems.

This Recommendation | International Standard is related to other Recommendations | International Standards in the set as defined by the Reference Model for Open Systems Interconnection. The Reference Model subdivides the area of standardization for interconnection into a series of layers of specification, each of manageable size.

The purpose of this Recommendation | International Standard is to define the service provided to the Presentation Layer at the boundary between the Session and Presentation Layers of the Reference Model. The session service is provided by the session protocol making use of the services available from the Transport Layer. This Recommendation | International Standard also defines the session service characteristics which the presentation protocol may exploit. The relationship between the Recommendations | International Standards for the session service, session protocol, transport service, and the presentation protocol is illustrated in Figure Intro.1 below.

It is recognized that, with respect to session Quality of Service (described in clause 10), work is still in progress to provide an integrated treatment of QOS across all of the layers of the OSI Reference Model and to ensure that the individual treatments in each layer service satisfy overall QOS objectives in a consistent manner. As a consequence, an addendum may be added to this Recommendation | International Standard at a later time which reflects further QOS developments and integration.

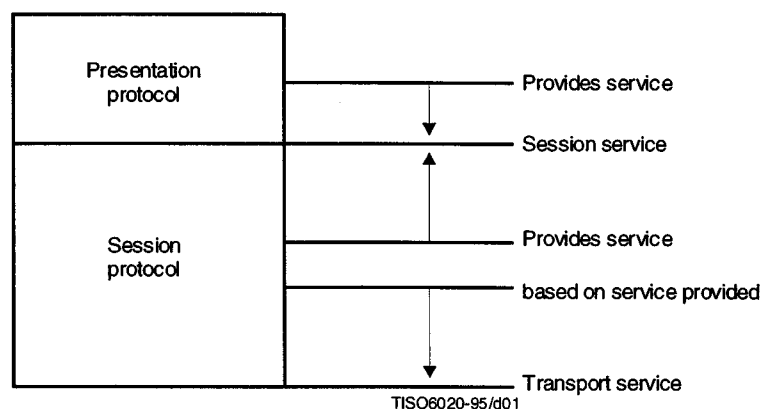


Figure Intro. 1 – Relationship of this Recommendation | International Standard to other OSI Standards

This page intentionally left blank

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION – SESSION SERVICE DEFINITION

SECTION 1 – GENERAL

1 Scope

This Recommendation | International Standard defines in an abstract way the externally visible service provided by the OSI Session Layer in terms of:

- a) the primitive actions and events of the service;
- b) the parameter data associated with each primitive action and event;
- c) the relationship between, and the valid sequence of these actions and events.

The service defined in this Recommendation | International Standard is that which is provided by the OSI session protocol (in conjunction with the transport service) and which may be used by the OSI presentation protocol.

This Recommendation | International Standard does not specify individual implementations or products, nor does it constrain the implementation of entities and interfaces within a computer system. There is, therefore, no conformance to this Recommendation | International Standard.

The text pertaining to the symmetric synchronization functional unit is not applicable to the support of ITU-T applications.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and International Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and International Standards listed below. Members of ISO and IEC maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*.
- ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: Conventions for the definition of OSI services*.
- ITU-T Recommendation X.214 (1993) | ISO/IEC 8072:1994, *Information technology – Open Systems Interconnection – Transport service definition*.
- ITU-T Recommendation X.225 (1995) | ISO/IEC 8327-1:1996, *Information technology – Open Systems Interconnection – Connection-oriented Session protocol: Protocol specification*.

- ITU-T Recommendation X.226 (1994) | ISO/IEC 8823-1:1994, *Information technology – Open Systems Interconnection – Connection-oriented presentation protocol: Protocol specification.*
- ITU-T Recommendation X.235 (1995) | ISO/IEC 9548-1:1995, *Information technology – Open Systems Interconnection – Connectionless Session protocol: Protocol specification.*

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.650 (1992), *Open Systems Interconnection (OSI) – Reference Model for naming and addressing.*
ISO 7498-3:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 3: Naming and addressing.*

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

NOTE – The definitions contained in this clause make use of abbreviations defined in clause 4.

3.1 Reference Model definitions

This Recommendation | International Standard is based on the concepts developed in ITU-T Rec. X.200 | ISO/IEC 7498-1, and makes use of the following terms defined in it:

- a) expedited-session-service-data-unit;
- b) session connection;
- c) Session Layer;
- d) session service;
- e) session-service-access-point;
- f) session-service-data-unit;
- g) Transport Layer;
- h) duplex;
- i) half-duplex.

3.2 Service convention definitions

This Recommendation | International Standard also makes use of the following terms defined in ITU-T Rec. X.210 | ISO/IEC 10731, as they apply to the Session Layer:

- a) service-user;
- b) service-provider;
- c) primitive;
- d) request;
- e) indication;
- f) response;
- g) confirm.

3.3 Session service definitions

3.3.1 calling SS-user: An SS-user that initiates a session connection establishment request.

3.3.2 called SS-user: An SS-user with whom a calling SS-user wishes to establish a session connection.

NOTE – Calling SS-users and called SS-users are defined with respect to a single connection. An SS-user can be both a calling and called SS-user simultaneously.

3.3.3 sending SS-user: An SS-user that acts as either a source of data during the data transfer phase of a session connection or a source of data during a particular instance of session-connectionless-mode transmission.

3.3.4 receiving SS-user: An SS-user that acts as either a sink of data during the data transfer phase of a session connection or a sink of data during a particular instance of session-connectionless-mode transmission.

NOTE – An SS-user can be both sending and a receiving SS-user simultaneously.

3.3.5 requestor; requesting SS-user: An SS-user that initiates a particular action.

3.3.6 acceptor; accepting SS-user: An SS-user that accepts a particular action.

3.3.7 token: An attribute of a session connection which is dynamically assigned to one SS-user at a time to permit certain services to be invoked.

3.3.8 conditional (parameter): A parameter whose presence in a request or response depends on conditions defined in the text of this Recommendation | International Standard; and whose presence in an indication or confirm is mandatory if that parameter was present in the preceding session service primitive, or absent if that parameter was absent in the preceding session service primitive.

3.3.9 proposed parameter: The value for a parameter proposed by an SS-user in an S-CONNECT request or an S CONNECT response that it wishes to use on the session connection.

3.3.10 selected parameter: The value for a parameter that has been chosen for use on the session connection.

3.3.11 session-connectionless-mode transmission; session-connectionless transmission: The transmission of a unit of data in a single self-contained session service without establishing a session connection (as defined in ITU-T Rec. X.200 | ISO/IEC 7498-1).

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply.

4.1 Data units

SSDU	Session service data unit
NSSDU	Normal data session service data unit
TSSDU	Typed data session service data unit
XSSDU	Expedited session service data unit

4.2 Miscellaneous

SS	Session service
SSAP	Session service access point
QOS	Quality of Service

4.3 Service variables

V(A)	See 11.4.1.1.1
V(M)	See 11.4.1.1.2
V(R)	See 11.4.1.1.3
Vsc	See 11.4.1.1.4

5 Conventions

This Recommendation | International Standard uses the descriptive conventions contained in the OSI Service Conventions (ITU-T Rec. X.210 | ISO/IEC 10731) except that, where indicated in this Recommendation | International Standard, parameter values associated with a service primitive may be passed in a direction opposite to the direction of the service primitive.

6 Model of the session service

6.1 Model of the layer service

This Recommendation | International Standard uses the abstract model for a layer service defined in ITU-T Rec. X.210 | ISO/IEC 10731. The model defines the interactions between the SS-users and the SS-providers which take place at the two SSAPs. Information to be passed between an SS-user and the SS-provider by service primitives, which may convey parameters.

There are two types of session service:

- a) A connection-mode service, which is defined in clause 2.
- b) A connectionless-mode service, which is defined in clause 3. The connectionless-mode service defines the feature given in 7.1 e). This service is invoked by the connectionless presentation P-UNIT-DATA service.

6.2 Model of session connectionless-mode transmission

This Recommendation | International Standard uses the abstract model of connectionless-mode transmission defined in ITU-T Rec. X.200 | ISO/IEC 7498-1. A defining characteristic of session connectionless-mode transmission is the independent nature of each invocation of the connectionless-mode session service.

7 Overview of the session service

7.1 General overview

The session service provides the means for organized and synchronized exchange of data between co-operating SS-users. It provides its users with means to:

- a) establish a connection with another SS-user, exchange data with that user in a synchronized manner, and release the connection in an orderly manner;
- b) negotiate for the use of tokens to exchange data, synchronize and release the connection, and to arrange for data exchange to be half-duplex or duplex;
- c) establish synchronization points within the dialogue and, in the event of errors, resume the dialogue from an agreed synchronization point;
- d) interrupt a dialogue and resume it later at a pre-arranged point.

In addition, the following service is described:

- e) a means by which a single unit of data is transmitted from one source SSAP to another SSAP in a single session service access, without first establishing or later releasing a session connection.

7.2 Token concept

A token is an attribute of a session connection which is dynamically assigned to one SS-user at a time to permit certain services to be invoked. It is the right to exclusive use of the service.

Four tokens are defined:

- a) the data token;
- b) the release token;
- c) the synchronize-minor token;
- d) the major/activity token.

A token is always in one of the following states:

- e) available, in which case it is always
 - 1) assigned to one SS-user, who then has the exclusive right to use the associated service (provided that no other restrictions apply); and
 - 2) not assigned to the other SS-user, who does not have the right to use the service but may acquire it later; or

- f) not available to either SS-user, in which case neither SS-user has the exclusive use of the associated service. The service then becomes inherently available to both SS-users (data transfer and release), or otherwise unavailable to both SS-users (synchronization and activities).

Restrictions related to the availability and assignment of tokens are defined in 11.2.

7.3 Synchronization and dialogue unit concepts

SS-users may insert synchronization points into the data they are transmitting. There are two methods to identify a synchronization point. A synchronization point can be identified by a single serial number if the right to send all synchronization points is token controlled. Alternatively, synchronization points can be identified by two serial numbers, one for each direction of flow. This dual numbering scheme, called symmetric synchronization, allows the SS-users to independently place synchronization points in their sending flows. Serial numbers are maintained by the SS-provider (see 11.4).

Any semantics which SS-users may give to their synchronization points are transparent to the SS-provider.

There are two types of synchronization points:

- a) minor synchronization points;
- b) major synchronization points.

Major synchronization points are used to structure the exchange of data into a series of dialogue units. The characteristic of a dialogue unit is that all communication within it is completely separated from all communication before and after it. A major synchronization point indicates the end of one dialogue unit and the beginning of the next. Each major synchronization point is confirmed explicitly.

Minor synchronization points are used to structure the exchange of data within a dialogue unit. Figure 1 illustrates how a dialogue unit is structured through the use of minor synchronization points. Each minor synchronization point may or may not be confirmed explicitly. A minor synchronization point inserted in association with the data separation service protects all data sent before the minor synchronization point from being discarded by a subsequent resynchronize request.

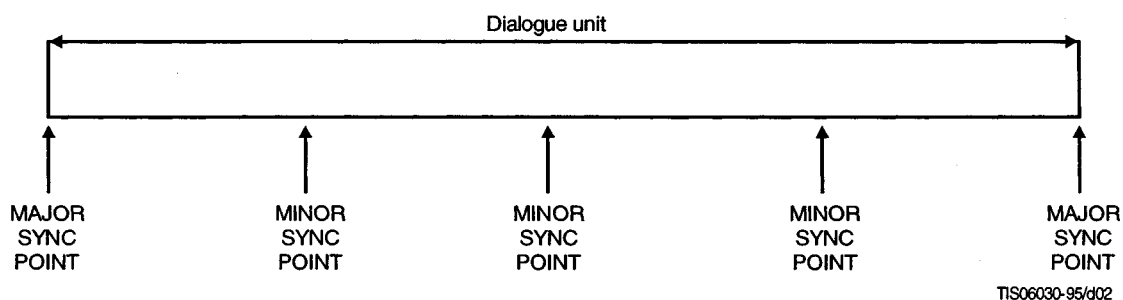


Figure 1 – Example of a structured dialogue unit

7.4 Activity concept

The activity concept allows SS-users to distinguish between different logical pieces of work called activities. Each activity consists of one or more dialogue units. Only one activity is allowed on a session connection at a time, but there may be several consecutive activities during a session connection. An activity may also span more than one session connection. An activity can be interrupted and then resumed on the same or on a subsequent session connection. This can be considered as a form of resynchronization.

Figure 2 shows how an activity may be structured into dialogue units through the use of major synchronization points. In addition, the SS-users may transfer data outside an activity.

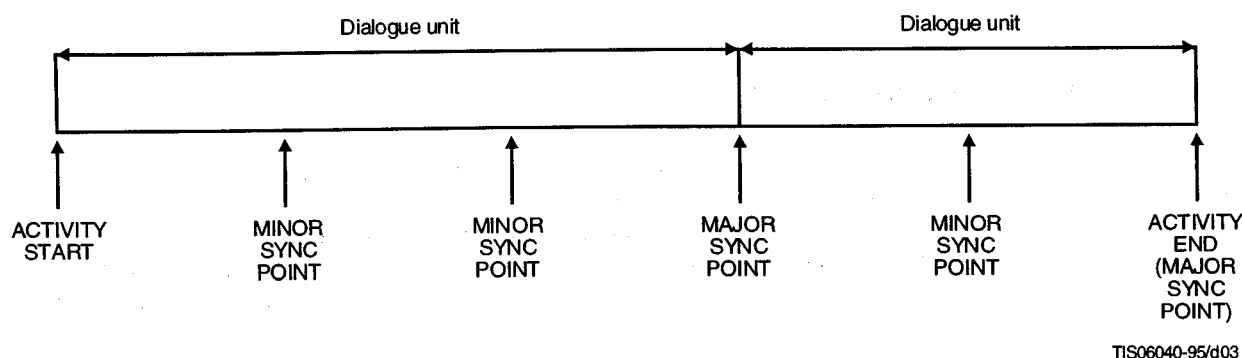


Figure 2 – Example of a structured activity

7.5 Resynchronization

Resynchronization may be initiated by either SS-user. It sets the session connection to a defined state, and therefore includes reassignment of tokens and setting the synchronization point serial number(s) to new value(s).

When symmetric synchronization is used, the SS-user can request to resynchronize one direction of data flow, or both directions. If both directions are requested, then both serial numbers are assigned new values; otherwise only the serial number associated with the requested direction of flow is assigned a new value. Resynchronization purges all undelivered data for the requested direction(s) of flow.

When symmetric synchronization is not used, both directions of flow are always resynchronized. The single synchronization point serial number is assigned a new value. All undelivered data is purged.

Three options are provided:

- abandon option which is used to set the synchronization point serial number to an unused value;
- restart option which is used to set the synchronization point serial number to any used value which is greater than the synchronization point serial number which identifies the last acknowledged major synchronization point;
- set option is used to set the synchronization point serial number to any value chosen by SS-user.

When symmetric synchronization is used, a resynchronize option (restart, set or abandon) is provided by the SS-user for each direction of data flow on which resynchronization is requested.

7.6 Negotiation

Negotiation takes place between both SS-users during the session connection establishment phase according to the following rules.

7.6.1 Negotiation of functional units

The kernel functional unit (see clause 9) is always used. Each SS-user proposes the use or non-use of each of the other functional units. A functional unit is selected only if both SS-users propose use of the functional unit and it is supported by the SS-provider. Specific negotiation rules are given in 12.1.2.

7.6.2 Negotiation of initial token settings

When the calling SS-user proposes use of a functional unit that requires a token, it also proposes the initial token settings:

- a) calling SS-user side;
- b) called SS-user side;
- c) called SS-user choice.

If the use of the functional unit is selected, the token is set to:

- d) the side proposed by the called SS-user, if "called SS-user choice" is proposed by the calling SS-user; or
- e) in all other cases, the side proposed by the calling SS-user.

7.6.3 Negotiation of initial synchronization point serial number

When a calling SS-user proposes any of the minor synchronize, symmetric synchronize, major synchronize or resynchronize functional units, but does not propose the activity functional unit, it also proposes values for the initial synchronization point serial number(s). Two serial numbers are proposed if the symmetric synchronize functional unit is proposed; otherwise one serial number is proposed. If two serial numbers are proposed, the First Initial Synchronization Point Serial Number is associated with the calling SS-user's sending flow and the Second Initial Synchronization Point Serial Number is associated with the calling SS-user's receiving flow.

The calling SS-user may also propose value(s) for the initial synchronization point serial number(s) even if the activity management functional unit is proposed provided that any of the minor synchronize, symmetric synchronize, major synchronize or resynchronize functional units are also proposed. If the called SS-user selects use of any of the minor synchronize, symmetric synchronize, major synchronize or resynchronize functional units, but does not select use of the activity management functional unit, it returns value(s) for the initial synchronization point serial number(s) which may or may not be the same as the values proposed by the calling SS-user. If the symmetric synchronize functional unit is selected, the called SS-user returns values for two serial numbers; otherwise a value for one serial number is returned. The values returned by the called SS-user are used as the initial synchronization point serial number(s) for the new session connection.

In all other combinations of functional units, no initial synchronization point serial number(s) are proposed.

8 Phases and services of the session service

The connection-mode session service comprises three phases. The purpose of each phase, and a short description of the associated services are given in 8.1 to 8.3. The services and the primitives by which they are invoked are defined in clauses 12, 13 and 14.

The connectionless-mode session service is used to transfer a single SSDU from the sending SS-user to the receiving SS-user. The service primitives associated with it are defined in clause 17.

NOTE – The amount of SS-user data which can be transferred in certain primitives may be limited due to protocol restrictions imposed by the SS-provider.

8.1 Session connection establishment phase

The session connection establishment phase is concerned with establishing a connection between two SS-users. It has one service associated with it:

The session connection service (see 12.1) is used to set up a session connection and to negotiate tokens and parameters to be used for the connection.

8.2 Data transfer phase

The data transfer phase is concerned with the exchange of data between the two SS-users connected in the session connection establishment phase.

There are four services associated with data transfer:

- a) The normal data transfer service (see 13.1) allows the transfer of normal data SSDUs (NSSDUs) over a session connection. Its use is controlled by the data token if the half-duplex functional unit has been selected.
- b) The expedited data transfer service (see 13.2) allows the transfer of expedited SSDUs (XSSDUs) over a session connection free from the token and flow control constraints of the normal data transfer service, typed data transfer service and capability data exchange service.
- c) The typed data transfer service (see 13.3) is used to transfer typed data SSDUs (TSSDUs) independent of the availability and assignment of the data token.
- d) The capability data exchange service (see 13.4) is used to exchange confirmed SS-user data while not within an activity.

There are three services concerned with token management:

- e) The give tokens service (see 13.5) allows an SS-user to surrender one or more specific tokens to the other SS-user.
- f) The please tokens service (see 13.6) allows an SS-user to request the other SS-user to transfer one or more specific tokens to it.
- g) The give control service (see 13.7) allows an SS-user to surrender all available tokens to the other SS-user.

There are five services associated with synchronization and resynchronization:

- h) The minor synchronization point service (see 13.8) allows the SS-user to separate the flow of NSSDUs and TSSDUs transmitted before the service was invoked from the subsequent flow of NSSDUs and TSSDUs. Its use is controlled by the synchronize-minor token.
- i) The symmetric synchronization point service (see 13.8) allows the SS-user to independently define minor synchronization points on each SS-user's sending flow. The synchronize-minor token is not available.
- j) The data separation service (see 13.8) adds functionality to the minor synchronization point service and the symmetric synchronize service to protect all data sent before the minor synchronization point from being discarded by a subsequent resynchronize request. There are no additional services associated with the data separation functional unit.
- k) The major synchronization point service (see 13.9) allows the SS-user to confine the flow of sequentially transmitted NSSDUs, TSSDUs and XSSDUs in each direction within a dialogue unit. Its use is controlled by the major/activity token.
- l) The resynchronize service (see 13.10) is used to set the session connection to a previous or to a new synchronization point and to reassign the available tokens. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.

There are two services for reporting errors or unanticipated situations:

- m) The provider-initiated exception reporting service (see 13.11) (P-exception reporting service) permits SS-users to be notified of exception conditions or SS-provider protocol errors. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.
- n) The user-initiated exception reporting service (see 13.12) (U-exception reporting service) is used by the SS-user to report an exception condition when the data token is available but not assigned to the SS-user. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.

There are five services associated with activities:

- o) The activity start service (see 13.13) is used to indicate that a new activity is entered. Its use is controlled by the major/activity token.
- p) The activity resume service (see 13.14) is used to indicate that a previously interrupted activity is re-entered. Its use is controlled by the major/activity token.
- q) The activity interrupt service (see 13.15) allows an activity to be abnormally terminated with the implication that the work so far achieved is not to be discarded and may be resumed later. Its use is controlled by the major/activity token. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.

- r) The activity discard service (see 13.16) allows an activity to be abnormally terminated with the implication that the work so far achieved is to be discarded, and not resumed. Its use is controlled by the major/activity token. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.
- s) The activity end service (see 13.17) is used to end an activity (and set a major synchronization point). Its use is controlled by the major/activity token.

Using the activity services may lead to a state where no activity is in progress on the session connection. When activity services are employed, but no activity is in progress, only the activity start, activity resume, token management, capability data, typed data, normal data, expedited data, abort and release services may be invoked by the SS-users.

8.3 Session connection release phase

The session connection release phase is concerned with releasing a previously established session connection. It has three services associated with it:

- a) The orderly release service (see 14.1) provides a means of achieving the orderly release of a session connection.
- b) The user-initiated abort service (see 14.2) (U-Abort service) is used to initiate the release of a session connection in a way that will terminate any outstanding service request. This service may cause loss of NSSDUs, TSSDUs and XSSDUs.
- c) The provider-initiated abort service (see 14.3) (P-Abort service) is used by the SS-provider to indicate the release of the session connection for internal reasons. This service may cause loss of NSSDUs, TSSDUs and XSSDUs. Any outstanding service request is terminated.

9 Functional units and subsets

9.1 Functional units

Functional units are logical groupings of related services defined by this Recommendation | International Standard for the purpose of:

- a) negotiation of SS-user requirements during the session connection establishment phase;
- b) reference by other Recommendation | International Standards.

Table 1 specifies the association of tokens and functional units. When a functional unit implies the availability of a token, services concerned with the management of that token are provided in order to be able to request and transfer the available tokens.

The services associated with each functional unit are specified in Table 2.

Table 1 – Functional units using tokens

Functional unit	Token
Negotiated release	Release token
Half-duplex	Data token
Minor synchronize	Synchronize-minor token
Major synchronize	Major/activity token
Activity management	Major/activity token

Table 2 – Services associated with each functional unit

Functional unit	Service(s)	Reference
Kernel (non-negotiable)	Session connection Normal data transfer Orderly release U-Abort P-Abort	12.1 13.1 14.1 14.2 14.3
Negotiated release	Orderly release Give tokens Please tokens	14.1 13.5 13.6
Half-duplex	Give tokens Please tokens	13.5 13.6
Duplex	No additional service	
Expedited data	Expedited data transfer	13.2
Typed data	Typed data transfer	13.3
Capability data	Capability data exchange	13.4
Minor synchronize	Minor synchronization point Give tokens Please tokens	13.8 13.5 13.6
Symmetric synchronize	Symmetric synchronize	13.8
Data separation	Data separation (associated with the minor synchronization point service or the symmetric synchronize service)	13.8
Major synchronize	Major synchronization point Give tokens Please tokens	13.9 13.5 13.6
Resynchronize	Resynchronize	13.10
Exceptions	Provider exception reporting User exception reporting	13.11 13.12
Activity management	Activity start Activity resume Activity interrupt Activity discard Activity end Give tokens Please tokens Give control	13.13 13.14 13.15 13.16 13.17 13.5 13.6 13.7

9.1.1 Kernel functional unit

The kernel functional unit supports the basic session services required to establish a session connection, transfer normal data and release the session connection.

9.1.2 Negotiated release functional unit

The negotiated release functional unit supports the negotiated orderly release service. The release token is available when this functional unit has been selected.

9.1.3 Half-duplex functional unit

The half-duplex functional unit supports the half-duplex service. The data token is available when this functional unit has been selected. It is not possible to select both this functional unit and the duplex functional unit for use on the same session connection.

9.1.4 Duplex functional unit

The duplex functional unit supports the duplex service. It is not possible to select both this functional unit and the half-duplex functional unit for use on the same session connection.

9.1.5 Expedited data functional unit

The expedited data functional unit supports the session expedited data transfer service.

9.1.6 Typed data functional unit

The typed data functional unit supports the typed data transfer service.

9.1.7 Capability data exchange functional unit

The capability data exchange functional unit supports the capability data exchange service. This functional unit can only be selected when the activity management functional unit has been selected.

9.1.8 Minor synchronize functional unit

The minor synchronize functional unit supports the minor synchronization point service. The synchronize-minor token is available when this functional unit has been selected.

9.1.9 Symmetric synchronize functional unit

The symmetric synchronize functional unit supports the symmetric synchronization point service which allows the SS-users to independently place minor synchronization points in their sending flows. Correlation of these synchronization points is an SS-user responsibility. The synchronize-minor token is not available. It is not possible to select this functional unit and the minor synchronize functional unit for use on the same session connection.

It is possible to select the symmetric synchronize functional unit with either the duplex functional unit or the half-duplex functional unit. If the half-duplex functional unit is selected, the data token is available; if the duplex functional unit is selected, the data token is not available. The synchronize-minor token is not available when the symmetric synchronize functional unit is selected.

9.1.10 Data separation functional unit

The data separation functional unit supports the data separation service and adds functionality to the minor synchronization point service and the symmetric synchronize service. The data separation functional unit can only be selected when either the minor synchronize functional unit has been selected or the symmetric synchronize functional unit has been selected. It is not valid to select both this functional unit and the activity management functional unit for use on the same session connection.

9.1.11 Major synchronize functional unit

The major synchronize functional unit supports the major synchronization point service. The major/activity token is available when this functional unit has been selected.

9.1.12 Resynchronize functional unit

The resynchronize functional unit supports the resynchronize service.

9.1.13 Exceptions functional unit

The exceptions functional unit supports the user exception and provider exception reporting services.

This functional unit can only be selected when the half-duplex functional unit has been selected.

9.1.14 Activity management functional unit

The activity management functional unit supports the activity management services and the give control service. The major/activity token is available when this functional unit has been selected.

9.2 Subsets

A subset is defined as a combination of the kernel functional unit together with any other set of functional units provided that:

- a) if the capability data functional unit is included in the subset, then the activity management functional unit is also included in the subset; and
- b) if the exceptions functional unit is included in the subset, then the half-duplex functional unit is also included in the subset;
- c) if the data separation functional unit is included in a subset, one of the minor synchronize functional unit and the symmetric synchronize functional unit is also included in the subset.

NOTE – This Recommendation | International Standard contains no requirements for the registration of subsets. Users of this Recommendation | International Standard may define subsets to meet their session service needs. Other Recommendations | International Standards may identify subsets that comply with the above definition.

10 Quality of session service

The term “Quality of Service” (QOS) refers to certain characteristics of a session connection as observed between the session connection endpoints. QOS describes aspects of a session connection which are attributable solely to the SS-provider; such aspects are independent of SS-user behaviour (which is beyond the control of the SS-provider). SS-user behaviour does not impact the QOS provided.

Once a session connection is established, the SS-users at the two ends have the same knowledge and understanding of what the QOS over the session connection is.

10.1 Determination of QOS

QOS is described in terms of QOS parameters.

The definition of the QOS parameters associated with the session service is given in 10.3. These definitions provide both SS-users and the SS-provider with a common understanding of the QOS characteristics.

Two types of session service QOS parameters are identified:

- a) Those which are negotiated during the session connection establishment phase:
 - 1) session connection protection (see 10.3.9);
 - 2) session connection priority (see 10.3.10);
 - 3) residual error rate (see 10.3.5);
 - 4) throughput, for each direction of transfer (see 10.3.3);
 - 5) transit delay, for each direction of transfer (see 10.3.4);
 - 6) optimized dialogue transfer (see 10.3.13); and
 - 7) extended control (see 10.3.12).
- b) Those which are not negotiated during the session connection establishment phase but whose values are selected and/or known by other methods (for example, *a priori* knowledge and agreement, or by means of management functions) not defined in this Recommendation | International Standard:
 - 1) session connection establishment delay (see 10.3.1);
 - 2) session connection establishment failure probability (see 10.3.2);
 - 3) transfer failure probability (see 10.3.6);
 - 4) session connection release delay (see 10.3.7);
 - 5) session connection release failure probability (see 10.3.8);
 - 6) session connection resilience (see 10.3.11).

The negotiation procedures for parameters listed in 10.1 a) are defined in 10.2. Once the session connection is established, the selected QOS parameters are not re-negotiated during the lifetime of the session connection. The SS-user should be aware that changes in QOS during a session connection are not signalled in the session service.

10.2 Session connection QOS negotiation procedures

QOS negotiation is described in terms of parameters which can be conveyed by the S-CONNECT primitives during the session connection establishment phase (see clause 12). For the parameters which are negotiated during the session connection establishment phase [see 10.1 a)], the parameter values and negotiation rules are defined as follows:

- a) In the S-CONNECT request primitive, the calling SS-user can specify:
 - 1) For session connection protection, session connection priority, extended control, and optimized dialogue transfer, a single parameter value which is the “desired” QOS; for extended control and optimized dialogue transfer, one of the two values “feature desired” or “feature not desired” is conveyed.

NOTE – If the calling SS-user proposes use of the expedited data functional unit, the extended control parameter has the value “feature desired”.
 - 2) For residual error rate, and for each direction of throughput and transit delay, two parameter values which are the “desired” QOS and the “lowest acceptable” QOS to which the calling SS-user will agree.
- b) In the S-CONNECT indication primitive, for each of the negotiated parameters, an “available” value is conveyed which is specified as follows:
 - 1) For session connection protection, if the SS-provider agrees to provide a QOS value equivalent to the “desired” value specified in the S-CONNECT request, then the SS-provider specifies that value as “available”; if the SS-provider does not agree to provide the “desired” QOS requested, the SS-provider refuses to establish the session connection by issuing the S-CONNECT (reject) confirm primitive to the calling SS-user.
 - 2) For session connection priority, the SS-provider specifies the QOS value it is willing to provide (a value which is equal to or better than the “desired” value specified in the S-CONNECT request) as “available”.
 - 3) For the residual error rate and each direction of throughput and transit delay, if the SS-provider agrees to provide a value of QOS which is equal to or better than the “lowest acceptable” QOS value specified in the S-CONNECT request, then the SS-provider specifies the value as “available”; if the SS-provider does not agree to provide this QOS, then the SS-provider refuses to establish the session connection by issuing the S-CONNECT (reject) confirm primitive to the calling SS-user.
 - 4) For extended control and optimized dialogue transfer, if the “desired” value in the S-CONNECT request primitive is “feature not desired” then “feature not desired” is specified as “available”; if the “desired” value is “feature desired” and the SS-provider agrees to provide the feature on the session connection, then “feature desired” is specified as “available”; otherwise if the SS-provider does not agree to provide the feature, “feature not desired” is specified as “available”.
- c) In the S-CONNECT response primitive, for each of the negotiated parameters, an “agreed” value is conveyed which is specified as follows:
 - 1) For optimized dialogue transfer, if the “available” value in the S-CONNECT indication primitive is “feature not desired” and the called SS-user agrees not to have the feature provided on the session connection, then “feature not desired” is specified as “agreed”; otherwise the SS-user may reject establishment of the session connection; if the “available” value in the indication primitive is “feature desired” and the SS-user agrees to have the feature provided, then “feature desired” is specified as “agreed”; otherwise, if the SS-user does not agree to provision of the feature, the value “feature not desired” is specified as “agreed”.
 - 2) For each of the other parameters, if the called SS-user agrees to the QOS value specified as “available” in the S-CONNECT indication primitive, then the identical value is specified as “agreed”; if the SS-user does not agree to the “available” value, the SS-user may reject establishment of the session connection.
- d) In the S-CONNECT confirm primitive, for each of the negotiated parameters, an “agreed” value is conveyed which is identical to the “agreed” value conveyed in the S-CONNECT response.

10.3 Definition of QOS parameters

QOS parameters can be classified as:

- a) parameters which express session service performance parameters, as shown in Table 3;
- b) parameters which express other session service characteristics, as shown in Table 4.

These session service QOS parameters are defined in this subclause.

Table 3 – Classification of Performance of QOS Parameters

Phase	Performance criterion	
	Speed	Accuracy/reliability
Session connection establishment	Session connection establishment delay	Session connection establishment failure probability (misconnection/session connection refusal)
Data transfer	Throughput Transit delay	Residual error rate (corruption) Session connection resilience Transfer failure probability
Session connection release	Session connection release delay	Session connection release failure probability

Table 4 – Parameters specifying other session service features

Extended Control
Session Connection Protection
Session Connection Priority
Optimized Dialogue Transfer

10.3.1 Session connection establishment delay

Session connection establishment delay is the maximum acceptable delay between an S-CONNECT request and the corresponding S-CONNECT confirm primitive.

NOTE – This delay includes SS-user dependent components.

10.3.2 Session connection establishment failure probability

Session connection establishment failure probability is the ratio of total session connection establishment failures to total session connection establishment attempts in a measurement sample.

Session connection establishment failure is defined to occur when a requested session connection is not established within the specified maximum acceptable session connection establishment delay as a result of misconnection, session connection refusal, or excessive delay on the part of the SS-provider. Session connection establishment attempts which fail as a result of error, session connection refusal, or excessive delay on the part of an SS-user are excluded in calculating session connection establishment failure probability.

10.3.3 Throughput

Throughput is defined for each direction of transfer, in terms of a sequence of at least two SSDUs successfully transferred by an S-DATA request/S-DATA indication or S-TYPED-DATA request/S-TYPED-DATA indication sequence of primitives. Given such a sequence of n SSDUs, where $n \geq 2$, the throughput is defined to be the smaller of:

- a) the number of SS-user data octets contained in the last $n - 1$ SSDUs divided by the time between the first and last S-DATA or S-TYPED-DATA request in the sequence; and
- b) the number of SS-user data octets contained in the last $n - 1$ SSDUs divided by the time between the first and the last S-DATA or S-TYPED-DATA indications in the sequence.

Successful transfer of the octets in a transmitted SSDU is defined to occur when the bits are delivered to the intended receiving SS-user without error, in the proper sequence, prior to release of the session connection by the receiving SS-user.

Throughput is only meaningful for a sequence of complete SSDUs and each specification is based on a previously stated average SSDU size.

Throughput is specified separately for each direction of transfer on a session connection. In each direction, a specification of throughput will consist of a "maximum throughput" value and an "average throughput" value. The "maximum throughput" value represents the maximum rate at which the SS-provider can continuously accept and deliver SSDUs, in the absence of sending SS-user input delays or flow control applied by the receiving SS-user. Thus, the sequence of SSDUs in the calculation above are defined to be presented continuously at the maximum rate. The "average throughput" value represents the expected transfer rate on a session connection including the effects of expected user-attributable delays (e.g. non-continuous SSDU input, receiving SS-user flow control). Thus, the sequence of SSDUs in the calculation above are defined to be presented at a rate which includes components representing "average" user delays.

It is possible for either the input or the output of a sequence of SSDUs to be excessively delayed by the SS-users. Such occurrences are excluded in calculating "average throughput" values.

For each direction of transfer, and for each of the "maximum throughput" and "average throughput" specifications, the throughput QOS for a particular session connection is negotiated between the SS-users and the SS-provider (see 10.2).

Throughput on a session connection relates only to the transfer of normal data and typed data over the session connection. There is no specification of the throughput for data which is transferred in association with the issue of any other session service primitives (e.g. S-CONNECT, S-CAPABILITY-DATA, etc.).

10.3.4 Transit delay

Transit delay is the elapsed time between the completion of any session service request primitive and the corresponding session service indication primitive occurring during the data transfer phase of a session connection. Elapsed time values are calculated only on service primitive pairs which are successfully completed.

Successful completion of a service primitive pair is defined to occur when the issue of the request primitive by one SS-user results in the issue of the corresponding indication primitive to the peer user (including any SS-user data associated with the primitive) which is without error, and in a proper sequence with respect to other primitives, prior to release of the session connection by the receiving SS-user.

In duplex and half-duplex session connections, transit delay is specified independently for each direction of transfer. In general, each transit delay specification defines both the average value and the maximum value expected for a session connection. Each specification of transit delay assumes a previously stated average size for SS-user data included in the service primitive pair.

An attempt to measure the transit delay for an individual service primitive pair may be greatly influenced if the receiving SS-user exercises flow control. Such occurrences are excluded in calculating both average and maximum transit delay values.

10.3.5 Residual error rate

Residual error rate is the ratio of total incorrect, lost, and duplicate units of SS-user data to the total units of SS-user data transferred across the session service boundary in association with any SS-primitive issued in the data transfer phase of a

session connection during a measurement period. The relationship between these quantities for a particular SS-user pair is defined in Figure 3.

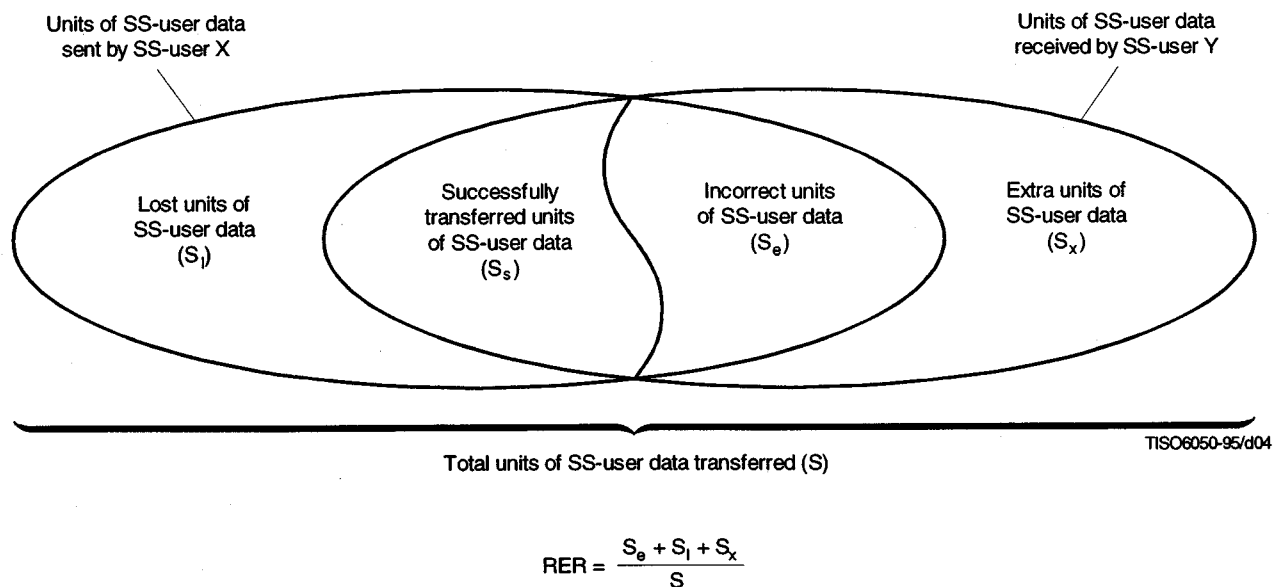


Figure 3 – Components of residual error rate

10.3.6 Transfer failure probability

Transfer failure probability is the ratio of total transfer failures to total transfer samples observed during a performance measurement.

A transfer sample is a discrete observation of SS-provider performance in handling service requests made by the SS-user. A transfer sample begins with the initiation of session service requests during the data transfer phase and continues until the outcome of a given number of service requests have been determined. These service requests may include the transfer of SS-user data or other service requests (such as S-ACTIVITY-START request, S-TOKEN-PLEASE request, etc.) made by the SS-user. A transfer sample will normally correspond to the duration of an individual session connection.

A transfer failure is a transfer sample in which the observed performance is worse than a specified minimum acceptable level. Transfer failures are identified by comparing the measured values for applicable supported performance parameters with specified transfer failure thresholds. The three supported performance parameters which may apply are throughput, transit delay, and residual error rate.

In systems where session service QOS is reliably monitored by the SS-provider, transfer failure probability can be estimated by the probability of an S-P-ABORT or an S-P-EXCEPTION-REPORT during a transfer sample.

10.3.7 Session connection release delay

Session connection release delay is the maximum acceptable delay between an SS-user initiated S-U-ABORT request and the successful release of a particular session connection. Session connection release delay is normally specified independently for each SS-user.

Issue of an S-U-ABORT request by either SS-user marks the beginning of the session connection release delay for both users. Successful release for one SS-user is defined to occur when that SS-user is first able to initiate a new session connection. Successful release is signalled to the SS-user not initiating the S-U-ABORT request by an S-U-ABORT indication. The SS-user initiating the S-U-ABORT request will normally receive a similar signal of local significance.

10.3.8 Session connection release failure probability

Session connection release failure probability is the ratio of total SS-user initiated abort requests resulting in session connection release failure to total SS-user initiated abort requests included in a measurement sample. Session connection release failure probability is normally specified independently for each SS-user.

Session connection release failure is defined to occur, for a particular SS-user, if that SS-user is not successfully released (as defined in 10.3.7) within the specified maximum session connection release delay as a result of error or excessive delay on the part of the SS-provider. Session connection release attempts which fail as a result of error, release refusal, or excessive delay on the part of an SS-user are excluded in calculating session connection release failure probability.

10.3.9 Session connection protection

Session connection protection is the extent to which an SS-provider attempts to prevent unauthorized monitoring or manipulation of SS-user originated information. Session connection protection is specified qualitatively by selecting one of the following session connection protection options:

- a) no protection features;
- b) protection against passive monitoring;
- c) protection against modification, replay, addition or deletion;
- d) both b) and c).

10.3.10 Priority

The specification of priority is concerned with the relationship between session connections. This parameter specifies the relative importance of a session connection with respect to:

- a) the order in which session connections are to have their QOS degraded, if necessary; and
- b) the order in which session connections are to be broken to recover resources, if necessary.

This parameter only has meaning in the context of some management entity or structure able to judge relative importance. The number of priority levels is limited.

10.3.11 Session connection resilience

Session connection resilience parameters specify the probability of:

- a) an SS-provider initiated non-orderly release of a session connection (i.e. issue of an S-P-ABORT indication); and
- b) an SS-provider exception report (i.e. issue of an S-P-EXCEPTION-REPORT indication) during a specified time interval on an established session connection.

10.3.12 Extended control parameter

The extended control parameter allows the SS-users to make use of the resynchronize, abort, activity interrupt and activity discard services when normal flow is congested.

NOTE – When the expedited data functional unit has been selected, the extended control QOS is always provided to the SS-users.

10.3.13 Optimized dialogue transfer

The optimized dialogue transfer QOS parameter permits the concatenated transfer of certain session service requests. How this concatenation of service requests is achieved is a local implementation matter.

NOTE – This QOS parameter invokes the SS-provider extended concatenation protocol option.

SECTION 2 – DEFINITION OF CONNECTION ORIENTED SESSION SERVICE PRIMITIVES

11 Introduction to session service primitives

11.1 Summary of primitives

Each of the services constituting the session service is achieved by invoking a sequence of session service primitives. Tables 5, 6 and 7 summarize the primitives and their parameters occurring in each phase of the session service. The parameters are defined in clauses 12, 13 and 14.

Table 5 – Session connection establishment phase primitives

Service	Primitives	Parameter
Session connection	S-CONNECT request S-CONNECT indication S-CONNECT response S-CONNECT confirm	Session Connection Identifier, Calling Session Address, Called Session Address, Responding Session Address, Result, QOS, Session Requirements, Synchronization Point Serial Number, Initial Assignment of Tokens, SS-user data

Table 6 – Data transfer phase primitives

Service	Primitives	Parameter
Normal data transfer	S-DATA request S-DATA indication	SS-user data
Expedited data transfer	S-EXPEDITED-DATA request S-EXPEDITED-DATA indication	SS-user data
Typed data transfer	S-TYPED-DATA request S-TYPED-DATA indication	SS-user data
Capability data exchange	S-CAPABILITY-DATA request S-CAPABILITY-DATA indication S-CAPABILITY-DATA response S-CAPABILITY-DATA confirm	SS-user data
Give tokens	S-TOKEN-GIVE request S-TOKEN-GIVE indication	Tokens, SS-user data
Please tokens	S-TOKEN-PLEASE request S-TOKEN-PLEASE indication	Tokens, SS-user data
Give control	S-CONTROL-GIVE request S-CONTROL-GIVE indication	SS-user data
Minor synchronization point	S-SYNC-MINOR request S-SYNC-MINOR indication S-SYNC-MINOR response S-SYNC-MINOR confirm	Type, Data separation, Synchronization point serial number, SS-user data
Major synchronization point	S-SYNC-MAJOR request S-SYNC-MAJOR indication S-SYNC-MAJOR response S-SYNC-MAJOR confirm	Synchronization point serial number, SS-user data
Resynchronize	S-RESYNCHRONIZE request S-RESYNCHRONIZE indication S-RESYNCHRONIZE response S-RESYNCHRONIZE confirm	Resynchronize type, Synchronization point serial number, Assignment of Tokens, SS-user data
P-exception report	S-P-EXCEPTION-REPORT indication	Reason
U-exception reporting	S-U-EXCEPTION-REPORT request S-U-EXCEPTION-REPORT indication	Reason, SS-user data
Activity start	S-ACTIVITY-START request S-ACTIVITY-START indication	Activity identifier, SS-user data
Activity resume	S-ACTIVITY-RESUME request S-ACTIVITY-RESUME indication	Activity identifier, Old activity identifier, Synchronization point serial number, Old session connection identifier, SS-user data
Activity interrupt	S-ACTIVITY-INTERRUPT request S-ACTIVITY-INTERRUPT indication S-ACTIVITY-INTERRUPT response S-ACTIVITY-INTERRUPT confirm	Reason, SS-user data
Activity discard	S-ACTIVITY-DISCARD request S-ACTIVITY-DISCARD indication S-ACTIVITY-DISCARD response S-ACTIVITY-DISCARD confirm	Reason, SS-user data
Activity end	S-ACTIVITY-END request S-ACTIVITY-END indication S-ACTIVITY-END response S-ACTIVITY-END confirm	Synchronization point serial number, SS-user data

Table 7 – Session connection release phase primitives

Service	Primitives	Parameter
Orderly release	S-RELEASE request S-RELEASE indication S-RELEASE response S-RELEASE confirm	Result, SS-user data
U-abort	S-U-ABORT request S-U-ABORT indication	SS-user data
P-abort	S-P-ABORT indication	Reason

11.2 Token restrictions on sending primitives

Table 8 defines the conditions under which those service primitives requiring tokens may be issued.

Table 8 – Token restrictions on service primitives

Service primitives	Data token	Synchronize minor token	Major/activity token	Release token
S-RELEASE request S-RELEASE response (negative)	2 nr	2 nr	2 nr	2 0
S-DATA request (half-duplex) S-DATA request (duplex)	1 3	nr nr	nr nr	nr nr
S-CAPABILITY-DATA request	2	2	1	nr
S-TOKEN-GIVE request (data token) S-TOKEN-GIVE request (sync-minor token) S-TOKEN-GIVE request (major/activity token) S-TOKEN-GIVE request (release token)	1 nr nr nr	nr 1 nr nr	nr nr 1 nr	nr nr nr 1
S-TOKEN-PLEASE request (data token) S-TOKEN-PLEASE request (sync-minor token) S-TOKEN-PLEASE request (major/activity token) S-TOKEN-PLEASE request (release token)	0 nr nr nr	nr 0 nr nr	nr nr 0 nr	nr nr nr 0
S-CONTROL-GIVE request	2	2	1	2
S-SYNC-MINOR request, without symmetric synchronization S-SYNC-MINOR request, with symmetric synchronization S-SYNC-MAJOR request	2 nr 2	1 3 2	nr nr 1	nr nr nr
S-U-EXCEPTION-REPORT request	0	nr	nr	nr
S-ACTIVITY-START request S-ACTIVITY-RESUME request S-ACTIVITY-INTERRUPT request S-ACTIVITY-DISCARD request S-ACTIVITY-END request	2 2 nr nr 2	2 2 nr nr 2	1 1 1 1 1	nr nr nr nr nr
0 Token available and not assigned to the SS-user who initiated the service primitive 1 Token available and assigned to the SS-user who initiated the service primitive 2 Token not available or token assigned to the SS-user who initiated the service primitive 3 Token not available nr No restriction				

11.3 Sequencing of primitives

All SS-user requests and responses are delivered by the SS-provider in the order in which they are submitted by the SS-user, except for the following:

- a) S-EXPEDITED-DATA;
- b) S-RESYNCHRONIZE;
- c) S-ACTIVITY-INTERRUPT;
- d) S-ACTIVITY-DISCARD;
- e) S-U-ABORT,

which may be delivered earlier than previously submitted primitives, but not later than subsequently submitted primitives.

11.4 Synchronization point serial number management

Certain primitives carry one or two synchronization point serial numbers to identify a synchronization point. The upper limit of a serial number is defined by the SS-users. It is referred to as the Upper Limit Serial Number and is in the range 1 000 000 to infinity. The Upper Limit Serial Number is negotiated during the session connection establishment phase. The means by which the Upper Limit Serial Number is passed between the SS-users and the SPM is a local matter.

Synchronization points are assigned valid synchronization point serial numbers in the range 0 to $(10^{**}\text{Upper Limit Serial Number}) - 2$ by the SS-provider. It is the responsibility of the SS-user to ensure that the number(s) assigned by the SS-provider in a synchronization point request do not exceed $(10^{**}\text{Upper Limit Serial Number}) - 2$.

The synchronization point serial number $(10^{**}\text{Upper Limit Serial Number}) - 1$ is also a valid synchronization point serial number for use by the SS-user, but only in the following services, which require the synchronization point serial number of the next synchronization point:

- a) Session Connection Service;
- b) Resynchronization Service.

The management of synchronization point serial numbers is defined in this Recommendation | International Standard in terms of:

- c) operations on abstract local variables $V(M)$, $V(A)$, $V(R)$ and V_{sc} , managed by the SS-provider; and
- d) primitives issued by the SS-user in order to invoke these operations.

These operations are summarized in Table A.4.

A synchronization point may be identified in either of the following two ways, depending on whether the symmetric synchronize functional unit has been selected:

- e) *Single serial number synchronization points*: Only one SS-user is allowed to define synchronization points at any given time, as controlled by the synchronize-minor and major/activity tokens. A single serial number is used to identify all synchronization points; it is incremented each time either SS-user issues a synchronization request. Single serial number synchronization points are used when the symmetric synchronize functional unit has not been selected. With this option, only the First Synchronization Point Serial Number parameter is used with the service primitives.
- f) *Symmetric serial number synchronization points*: Both SS-users are allowed to define minor synchronization points on their sending directions of data flow. Correlation of two minor synchronization points (one for each direction of flow) is an SS-user responsibility. The synchronize-minor token is not available. Only the serial number associated with the direction of flow on which a minor synchronization point travels is incremented. Symmetric serial number synchronization points are used when the symmetric synchronize functional unit has been selected. With this option, both the First and Second Synchronization Point Serial Number parameters are used with the service primitives. The First Synchronization Point Serial Number is used to indicate points on the requesting SS-user's sending flow; the Second Synchronization Point Serial Number is used to indicate points on the opposite direction of flow.

With symmetric serial number synchronization points, a minor synchronization point is associated with a single direction of data flow and can be represented by one synchronization point serial number. Major synchronization points and the ends of activities are always defined by two correlated synchronization point serial numbers, one for each direction of data flow.

When the data token is available and the symmetric synchronize functional unit has been selected, ownership of the data token is not required to issue a minor synchronization point. This permits a half-duplex user to indicate minor synchronization points in its typed-data flow against the turn, when the typed data functional unit is selected. If an SS-user using half-duplex mode does not wish to synchronize the typed data flow, the SS-user should restrict the right to issue minor synchronization points to the owner of the data token.

When both the major synchronize and symmetric synchronize functional units are selected, the major/activity token is available. Acknowledgement of a major synchronization defines a serial number for the direction of flow on which the confirmation travels. Only the owner of the major/activity token (and the data token, if available) is allowed to make an S-SYNC-MAJOR request.

For the case when the symmetric synchronize functional unit has not been selected, 11.4.1 defines management of single serial synchronization point numbers. For the case when the symmetric synchronize functional unit has been selected, 11.4.2 defines management of symmetric serial number synchronization points.

11.4.1 Management of single serial synchronization point numbers

This subclause defines the synchronization variables and how they are managed when the symmetric synchronize functional unit has not been selected for use on the session connection.

11.4.1.1 Variables

The synchronization variables $V(A)$, $V(M)$, and $V(R)$ are each single number variables when the symmetric synchronize functional unit has not been selected.

11.4.1.1.1 $V(A)$

$V(A)$ is the lowest serial number to which a synchronization point confirmation is expected. No confirmation is expected when $V(A) = V(M)$.

11.4.1.1.2 $V(M)$

$V(M)$ is the next serial number to be used.

11.4.1.1.3 $V(R)$

$V(R)$ is the lowest serial number to which resynchronization restart is permitted.

11.4.1.1.4 V_{sc}

V_{sc} is used to determine whether or not the SS-user has the right to send minor synchronization point responses. V_{sc} has the following values:

$V_{sc} = \text{true}$: the SS-user has the right to issue minor synchronization point responses when $V(A)$ is less than $V(M)$;

$V_{sc} = \text{false}$: the SS-user does not have the right to issue minor synchronization point responses.

11.4.1.2 Session connection establishment

When a session connection is established in which the symmetric synchronize functional unit has not been selected and at least one of the following functional units has been selected:

- a) minor synchronize functional unit; or
- b) major synchronize functional unit; or
- c) resynchronize functional unit

and the activity management functional unit has not been selected, $V(M)$ and $V(A)$ are set to the initial synchronization point serial number of the response/confirm primitives. $V(R)$ is set to zero. V_{sc} is set false.

11.4.1.3 Minor synchronization point

When the minor synchronize functional unit has been selected without the symmetric synchronize functional unit, the minor synchronize service has the following effect on synchronization variables.