

---

British Standard for

# Open Systems Interconnection: Basic connection-oriented session protocol specification

[ISO title: Information processing systems — Open Systems Interconnection —  
Basic connection oriented session protocol specification]

---

Interconnexion des Systèmes Ouverts: Protocole de session en mode connexion

Kommunikation Offener Systeme; Spezifikation des verbindungsorientierten  
Basis-Protokolls der Kommunikationssteuerungsschicht

## Contents

|                                                    | Page       |                                                  | Page |
|----------------------------------------------------|------------|--------------------------------------------------|------|
| National foreword                                  | 1          | 22 Parameters of the CAPABILITY DATA             |      |
| Committees responsible                             | Back cover | ACK SPDU                                         | 45   |
| <b>Specification</b>                               |            | 23 Parameters of the GIVE TOKENS SPDU            | 46   |
| 0 Introduction                                     | 2          | 24 Parameters of the PLEASE TOKENS SPDU          | 46   |
| 1 Scope and field of application                   | 3          | 25 Parameters of the MINOR SYNC POINT SPDU       | 47   |
| 2 References                                       | 3          | 26 Parameters of the MINOR SYNC ACK SPDU         | 47   |
|                                                    |            | 27 Parameters of the MAJOR SYNC POINT SPDU       | 48   |
| <b>Section one. General</b>                        |            | 28 Parameters of the MAJOR SYNC ACK SPDU         | 48   |
| 3 Definitions                                      | 4          | 29 Parameters of the RESYNCHRONIZE SPDU          | 49   |
| 4 Symbols and abbreviations                        | 5          | 30 Parameters of the RESYNCHRONIZE               |      |
| 5 Overview of the session protocol                 | 5          | ACK SPDU                                         | 49   |
|                                                    |            | 31 Parameters of the PREPARE SPDU                | 50   |
| <b>Section two. Session protocol specification</b> |            | 32 Parameters of the EXCEPTION REPORT SPDU       | 50   |
| 6 Use of the transport service                     | 15         | 33 Parameters of the EXCEPTION DATA SPDU         | 51   |
| 7 Elements of procedure related to SPDUs           | 20         | 34 Parameters of the ACTIVITY START SPDU         | 51   |
| 8 Structure and encoding of SPDUs                  | 34         | 35 Parameters of the ACTIVITY RESUME SPDU        | 52   |
|                                                    |            | 36 Parameters of the ACTIVITY INTERRUPT          |      |
| <b>Section three. Conformance</b>                  |            | SPDU                                             | 52   |
| 9 Conformance to this International Standard       | 54         | 37 Parameters of the ACTIVITY DISCARD SPDU       | 53   |
|                                                    |            | 38 Parameters of the ACTIVITY END SPDU           | 53   |
| <b>Annexes</b>                                     |            | 39 Incoming events                               | 60   |
| A State tables                                     | 55         | 40 States                                        | 62   |
| B Relationship to CCITT Recommendation T.62        | 88         | 41 Outgoing events                               | 63   |
| encoding                                           |            | 42 Operations on variables                       | 65   |
| C PGIs and PIs reserved for use by CCITT           | 91         | 43 Specific actions                              | 66   |
| Recommendation T.62                                | 92         | 44 Predicates                                    | 67   |
| D Corrections incorporated in ISO 8327 : 1987      |            | 45 Connection establishment state table          | 68   |
|                                                    |            | 46 Data transfer state table                     | 69   |
| <b>Tables</b>                                      |            | 47 Synchronization state table                   | 70   |
| 1 Session service primitives                       | 7          | 48 Resynchronization state table                 | 73   |
| 2 Transport service primitives                     | 8          | 49 Activity interrupt and discard state table    | 77   |
| 3 Functional units                                 | 11         | 50 Activity start, resume and capability data    |      |
| 4 Tokens associated with functional units          | 12         | state table                                      | 79   |
| 5 Token restrictions                               | 12         | 51 Token management and exceptions state table   | 80   |
| 6 Category 0, 1 and 2 SPDUs                        | 18         | 52 Connection release state table                | 83   |
| 7 Valid basic concatenation of SPDUs               | 18         | 53 Abort state table                             | 85   |
| 8 Valid extended concatenation of SPDUs            | 19         | 54 Relationship between T.62 commands and        |      |
| 9 Contention winner                                | 30         | responses and ISO 8327 SPDUs                     | 88   |
| 10 SPDUs associated with the PREPARE SPDU          | 31         | 55 Relationship between T.62 PGI/PI and ISO 8327 |      |
| 11 Parameters of the CONNECT SPDU                  | 38         | parameters                                       | 89   |
| 12 Parameters of the ACCEPT SPDU                   | 40         | 56 PGIs and PIs reserved for use by CCITT        |      |
| 13 Parameters of the REFUSE SPDU                   | 42         | Recommendation T.62                              | 91   |
| 14 Parameters of the FINISH SPDU                   | 42         | <b>Figures</b>                                   |      |
| 15 Parameters of the DISCONNECT SPDU               | 43         | 1 Relationship between the session protocol      |      |
| 16 Parameters of the NOT FINISHED SPDU             | 43         | and adjacent services                            | 2    |
| 17 Parameters of the ABORT SPDU                    | 43         | 2 Model of the Session Layer                     | 6    |
| 18 Parameters of the DATA TRANSFER SPDU            | 44         | 3 Illustration of TSDU structures                | 17   |
| 19 Parameters of the EXPEDITED SPDU                | 44         | 4 Illustration of definition of TSDU structure   | 35   |
| 20 Parameters of the TYPED DATA SPDU               | 45         | 5 Illustration of structure of SPDUs, PGI and    |      |
| 21 Parameters of the CAPABILITY DATA SPDU          | 45         | PI units                                         | 35   |
|                                                    |            | 6 Examples of SPDU structure                     | 36   |

## National foreword

This British Standard has been prepared under the direction of the Information Systems Technology Standards Committee. It is identical with ISO 8327 : 1987 'Information processing systems — Open Systems Interconnection — Basic connection oriented session protocol specification', published by the International Organization for Standardization (ISO). It is a revision of DD 112 : 1985, which is withdrawn.

The UK has actively participated through BSI Technical Committee IST/21 in the development of the international standard.

The text for the international standard has been developed in conjunction with The International Telegraph and Telephone Consultative Committee (CCITT), and will be technically identical to CCITT Recommendation X.225 : 1988, which will be available from BSI Sales Department, Linford Wood, Milton Keynes MK14 6LE.

Draft addenda to the international standard covering session symmetric synchronization, unlimited user data and connectionless extensions are in preparation.

This British Standard is closely related to BS 6960, which defines the service provided by the protocol specified in this British Standard.

Up-to-date information on existing and projected BSI publications for Open Systems Interconnection is included in the 'Basic List', published as part of the BSI Information Technology Services (BITS) initiative. For details contact: Marketing Department, BSI, Linford Wood, Milton Keynes MK14 6LE.

**Terminology and conventions.** The text of the international standard has been approved as suitable for publication as a British Standard without deviation. Some terminology and certain conventions are not identical with those used in British Standards; attention is drawn especially to the following.

Wherever the words 'International Standard' appear, referring to this standard, they should be read as 'British Standard'.

### Cross-references

| International standard | Corresponding British Standard                                                                                                                                 |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ISO 7498 : 1984        | BS 6568 Reference model of Open Systems Interconnection<br>Part 1 : 1988 Basic reference model (incorporating connectionless-mode transmission)<br>(Identical) |
| ISO 8326 : 1987        | BS 6960 : 1988 Open Systems Interconnection. Basic connection-oriented session service<br>definition<br>(Identical)                                            |

The Technical Committee has reviewed the provisions of ISO/DIS 7498-3 (referred to as ISO 7498/Add. 3) and ISO 8072, to which reference is made in the text, and has decided that they are acceptable for use in conjunction with this standard. A British Standard identical with ISO 8072 is in preparation.

**Compliance with a British Standard does not of itself confer immunity from legal obligations.**

British Standard for

# Open Systems Interconnection: Basic connection-oriented session protocol specification

## 0 Introduction

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

This International Standard is related to other International Standards in the set as defined by the Reference Model for Open Systems Interconnection (ISO 7498). The Reference Model subdivides the area of standardization for interconnection into a series of layers of specification, each of manageable size. It is most closely related to and lies within the field of application of the Session Service Definition (ISO 8326). It also uses and references the Transport Service Definition (ISO 8072), whose provisions it assumes in order to accomplish the aims of the session protocol. The interrelationship of these International Standards is illustrated in figure 1.

This International Standard specifies a single protocol with a common encoding.

It is intended that the session protocol should be general enough to cater for the total range of session service users without restricting future extensions.

The protocol is structured so that subsets of protocol can be defined.

The primary aim of this International Standard is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer session entities at the time of communication. These rules for communication are intended to provide a sound basis for development in order to serve a variety of purposes :

- a) as a guide for implementors and designers;
- b) for use in the testing and procurement of equipment;

- c) as part of an agreement for the admittance of systems into the open systems environment;
- d) as a refinement to the understanding of OSI.

As it is expected that the initial users of this International Standard will be designers and implementors of equipment this International Standard contains, in notes or in annexes, guidance on the implementation of the procedures defined herein.

It should be noted that, as the number of valid protocol sequences is very large, it is not possible with current technology to verify that an implementation will operate the protocol defined in this International Standard correctly under all circumstances. It is possible by means of testing to establish confidence that an implementation correctly operates the protocol in a representative sample of circumstances. It is, however, intended that this International Standard can be used in circumstances where two implementations fail to communicate in order to determine whether one or both have failed to operate the protocol correctly.

The variations and options available within this International Standard are essential as they enable a session service to be provided for a wide variety of applications. Thus, a minimally conforming implementation will not be suitable for use in all possible circumstances. It is important, therefore, to qualify all references to this International Standard with statements of the options provided or required or with statements of the intended purpose of provision or use.

This International Standard contains the following annexes :

- a) annex A — State tables;
- b) annex B — Relationship to CCITT Recommendation T.62 encoding;

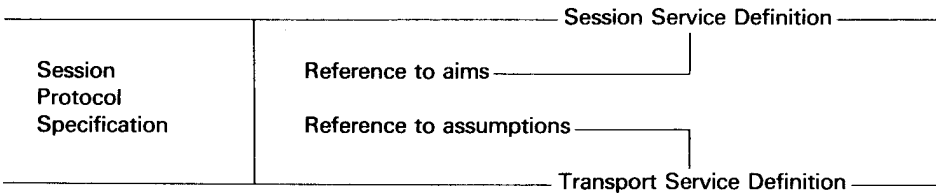


Figure 1 — Relationship between the session protocol and adjacent services

- c) annex C — PGIs and Pls reserved for use by Recommendation T.62.

## 1 Scope and field of application

This International Standard specifies

- a) procedures for a single protocol for the transfer of data and control information from one session entity to a peer session entity;
- b) the means of selecting the functional units to be used by the session entities;
- c) the structure and encoding of the session protocol data units used for the transfer of data and control information.

The procedures are defined in terms of

- a) the interactions between peer session entities through the exchange of session protocol data units;
- b) the interactions between a session entity and the session service user in the same system through the exchange of session service primitives;
- c) the interactions between a session entity and the transport service provider through the exchange of transport service primitives.

These procedures are applicable to instances of communication between systems which support the Session Layer of the OSI Reference Model and which wish to interconnect in an open systems environment.

This International Standard also specifies conformance requirements for systems implementing these procedures. It does not contain tests which can be used to demonstrate this conformance.

## 2 References

ISO 7498, *Information processing systems — Open Systems Interconnection — Basic Reference Model*.

ISO 7498/Add. 3, *Information processing systems — Open Systems Interconnection — Basic Reference Model — Addendum 3: Naming including addressing*.<sup>1)</sup>

ISO 8072, *Information processing systems — Open Systems Interconnection — Transport Service Definition*.

ISO 8326, *Information processing systems — Open Systems Interconnection — Basic connection oriented session service definition*.

CCITT Recommendation T.62, *Control Procedures for the Teletex and Group 4 Facsimile Services*.

NOTE — CCITT Recommendation T.62 is not essential for the application of this International Standard, but is included in the list of references as it has been referred to, for information, in relation to interworking with the CCITT Telematic services (see annex B and C).

<sup>1)</sup> At present at the stage of draft; publication anticipated in due course.

## Section one : General

### 3 Definitions

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

**3.1** This International Standard is based on the concepts developed in ISO 7498, and makes use of the following terms defined in it :

- a) expedited-session-service-data-unit;
- b) session-connection;
- c) Session Layer;
- d) session-protocol-data-unit;
- e) session-service;
- f) session-service-access-point;
- g) session-service-data-unit;
- h) Transport Layer;
- i) transport-connection;
- j) transport-service;
- k) transport-service-access-point;
- l) concatenation;
- m) segmenting;
- n) session selector (defined in ISO 7498/Add. 3).

**3.2** This International Standard is also based on concepts developed in ISO 8326 and makes use of the following terms defined in it :

- a) token;
- b) calling SS-user;
- c) called SS-user;
- d) sending SS-user;
- e) receiving SS-user;
- f) requesting SS-user;
- g) accepting SS-user;
- h) requestor;
- i) acceptor.

NOTE — The following terms used in this International Standard are used in relation to tokens and are explained in ISO 8326 :

- a) assigned;
- b) not assigned;
- c) available;
- d) not available.

**3.3** For the purposes of this International Standard, the following definitions also apply.

**3.3.1 Session Protocol Machine (SPM)** : An abstract machine that carries out the procedures specified in this protocol.

NOTE — A session entity comprises one or more SPMs.

**3.3.2 session-service user (SS-user)** : An abstract representation of the totality of those entities within a single system that make use of the session service.

**3.3.3 transport-service provider (TS-provider)** : An abstract machine which models the totality of the entities providing the transport service, as viewed by a session entity.

**3.3.4 local matter** : A decision made by a system concerning its behaviour in the Session Layer that is not subject to the requirements of this protocol.

**3.3.5 initiator** : An SPM that initiates a CONNECT SPDU.

**3.3.6 responder** : An SPM with whom an initiator wishes to establish a session connection.

NOTE — Initiator and responder are defined with respect to a single session connection.

**3.3.7 sending SPM** : An SPM that sends a given SPDU.

**3.3.8 receiving SPM** : An SPM that receives a given SPDU.

**3.3.9 owner (of a token)** : The SPM to whom a token is assigned.

**3.3.10 proposed parameter** : The value for a parameter proposed by an SPM in a CONNECT SPDU or an ACCEPT SPDU that it wishes to use on the session connection.

**3.3.11 negotiation** : The process by which two SPMs agree on a common set of functional units and protocol values and on the initial setting of available tokens.

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

**3.3.13 valid SPDU** : An SPDU which complies with the requirements of this International Standard with respect to structure and encoding.

**3.3.14 invalid SPDU** : An SPDU which does not comply with the requirements of this International Standard with respect to structure and encoding.

**3.3.15 protocol error** : Use of an SPDU that does not comply with the procedures agreed for the session connection.

**3.3.16 transparent (data)** : SS-user data which is transferred intact between SPMs and which is unavailable for use by the SPMs.

**3.3.17 SPDU identifier (SI) :** Heading information that identifies the SPDU concerned.

**3.3.18 length indicator (LI) :** An indicator that represents the length of an associated parameter field.

**3.3.19 parameter field :** A group of one or more octets used to represent a particular set of information.

**3.3.20 parameter identifier (PI) :** An identifier, defined in this International Standard, that indicates the type of information contained in its associated parameter field.

**3.3.21 PI unit :** An element of an SPDU that contains a PI field together with its associated LI field and parameter field.

**3.3.22 parameter group identifier (PGI) :** An identifier, defined in this International Standard, that indicates the type of information contained in its associated parameter field. The associated parameter field may consist of a set of PI units.

**3.3.23 PGI unit :** An element of an SPDU that contains a PGI field together with its associated LI field and parameter field.

**3.3.24 parameter value (PV) :** Information that represents the value of the parameter identified by either a PI or PGI.

**3.3.25 local variable :** A local variable within the SPM which is used as a means of clarifying the effects of certain actions and clarifying the conditions under which certain actions are permitted.

## 4 Symbols and abbreviations

### 4.1 Data units

SPDU session-protocol-data-unit  
SSDU session-service-data-unit  
TSDU transport-service-data-unit

### 4.2 SPDU fields

SI SPDU identifier (see 3.3.17)  
LI length indicator (see 3.3.18)  
PI parameter identifier (see 3.3.20)  
PGI parameter group identifier (see 3.3.22)  
PV parameter value (see 3.3.24)

### 4.3 Timer variables

TIM Disconnection and abort timer

## 4.4 Miscellaneous

SPM Session Protocol Machine (see 3.3.1)  
SS session-service  
SSAP session-service-access-point  
TSAP transport-service-access-point

## 4.5 Local variables

Vact See 5.8.1  
Vnextact See 5.8.2  
V(A) See 5.8.3  
V(M) See 5.8.4  
V(R) See 5.8.5  
Vsc See 5.8.6

## 5 Overview of the session protocol

### 5.1 Model of the Session Layer

The SPM (see the note) within the Session Layer communicates with the SS-user through an SSAP by means of the service primitives as defined by the session service definition (ISO 8326). Service primitives will cause or be the result of session protocol data unit exchanges between the peer SPMs using a transport connection. These protocol exchanges are effected using the services of the Transport Layer as defined by the transport service definition (ISO 8072) through two TSAPs.

Session connection endpoints are identified in end systems by an internal, implementation dependent, mechanism so that the SS-user and the SPM can refer to each session connection.

The model of the Session Layer is illustrated in figure 2.

NOTE — A session entity comprises one or more SPMs.

### 5.2 Services provided by the Session Layer

The protocol specified in this International Standard supports the session service defined in ISO 8326. Information is transferred to and from the SS-user using the session service primitives listed in table 1. Table 1 also defines the SPDUs associated with each of the service primitives.

### 5.3 Services assumed from the Transport Layer

The protocol specified in this International Standard assumes the use of the connection-oriented transport service defined in ISO 8072.

Information is transferred to and from the TS-provider in the transport service primitives listed in table 2.

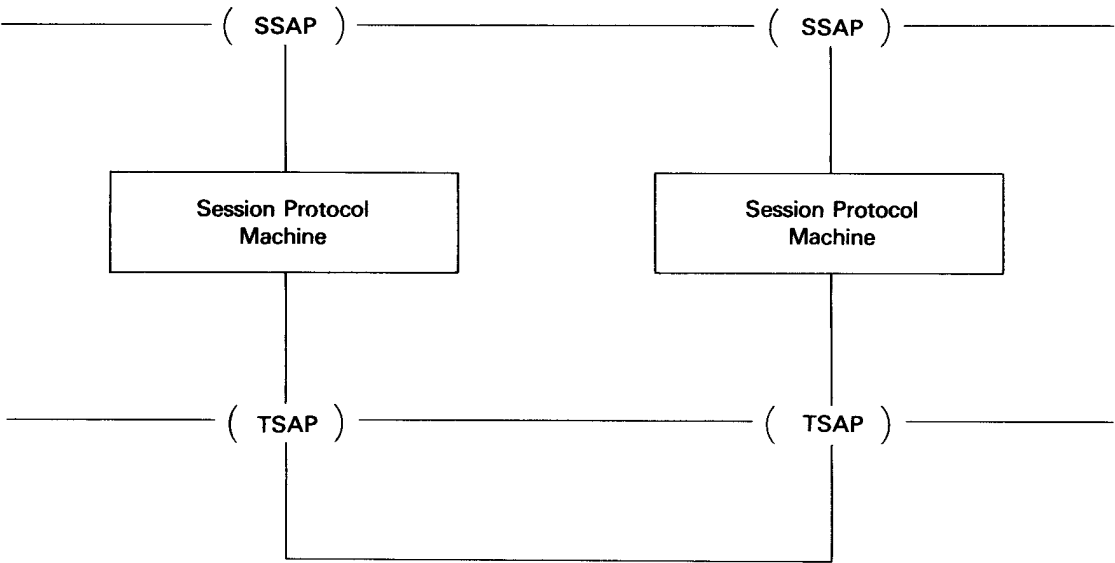


Figure 2 — Model of the Session Layer

Table 1 — Session service primitives

| Service                     | Primitives                                                                                                                                                          | Associated SPDUs                                                                                     |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| Session connection          | S-CONNECT request<br>S-CONNECT indication<br>S-CONNECT (accept) response<br>S-CONNECT (accept) confirm<br>S-CONNECT (reject) response<br>S-CONNECT (reject) confirm | CONNECT SPDU<br>CONNECT SPDU<br>ACCEPT SPDU<br>ACCEPT SPDU<br>REFUSE SPDU<br>REFUSE SPDU             |
| Normal data transfer        | S-DATA request<br>S-DATA indication                                                                                                                                 | DATA TRANSFER SPDU<br>DATA TRANSFER SPDU                                                             |
| Expedited data transfer     | S-EXPEDITED-DATA request<br>S-EXPEDITED-DATA indication                                                                                                             | EXPEDITED DATA SPDU<br>EXPEDITED DATA SPDU                                                           |
| Typed data transfer         | S-TYPED-DATA request<br>S-TYPED-DATA indication                                                                                                                     | TYPED DATA SPDU<br>TYPED DATA SPDU                                                                   |
| Capability data exchange    | S-CAPABILITY-DATA request<br>S-CAPABILITY-DATA indication<br>S-CAPABILITY-DATA response<br>S-CAPABILITY-DATA confirm                                                | CAPABILITY DATA SPDU<br>CAPABILITY DATA SPDU<br>CAPABILITY DATA ACK SPDU<br>CAPABILITY DATA ACK SPDU |
| Give tokens                 | S-TOKEN-GIVE request<br>S-TOKEN-GIVE indication                                                                                                                     | GIVE TOKENS SPDU<br>GIVE TOKENS SPDU                                                                 |
| Please tokens               | S-TOKEN-PLEASE request<br>S-TOKEN-PLEASE indication                                                                                                                 | PLEASE TOKENS SPDU<br>PLEASE TOKENS SPDU                                                             |
| Give control                | S-CONTROL-GIVE request<br>S-CONTROL-GIVE indication                                                                                                                 | GIVE TOKENS CONFIRM SPDU<br>GIVE TOKENS CONFIRM SPDU                                                 |
| Minor synchronization point | S-SYNC-MINOR request<br>S-SYNC-MINOR indication<br>S-SYNC-MINOR response<br>S-SYNC-MINOR confirm                                                                    | MINOR SYNC POINT SPDU<br>MINOR SYNC POINT SPDU<br>MINOR SYNC ACK SPDU<br>MINOR SYNC ACK SPDU         |
| Major synchronization point | S-SYNC-MAJOR request<br>S-SYNC-MAJOR indication<br>S-SYNC-MAJOR response<br>S-SYNC-MAJOR confirm                                                                    | MAJOR SYNC SPDU<br>MAJOR SYNC POINT SPDU<br>MAJOR SYNC ACK SPDU<br>MAJOR SYNC ACK SPDU               |
| Resynchronization           | S-RESYNCHRONIZE request<br>S-RESYNCHRONIZE indication<br>S-RESYNCHRONIZE response<br>S-RESYNCHRONIZE confirm                                                        | RESYNCHRONIZE SPDU<br>RESYNCHRONIZE SPDU<br>RESYNCHRONIZE ACK SPDU<br>RESYNCHRONIZE ACK SPDU         |
| P-Exception report          | S-P-EXCEPTION-REPORT indication                                                                                                                                     | EXCEPTION REPORT SPDU                                                                                |
| U-Exception reporting       | S-U-EXCEPTION-REPORT request<br>S-U-EXCEPTION-REPORT indication                                                                                                     | EXCEPTION DATA SPDU<br>EXCEPTION DATA SPDU                                                           |
| Activity start              | S-ACTIVITY-START request<br>S-ACTIVITY-START indication                                                                                                             | ACTIVITY START SPDU<br>ACTIVITY START SPDU                                                           |

Table 1 (concluded)

| Service            | Primitives                                                                                                                                                          | Associated SPDUs                                                                                                 |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| Activity resume    | S-ACTIVITY-RESUME request<br>S-ACTIVITY-RESUME indication                                                                                                           | ACTIVITY RESUME SPDU<br>ACTIVITY RESUME SPDU                                                                     |
| Activity interrupt | S-ACTIVITY-INTERRUPT request<br>S-ACTIVITY-INTERRUPT indication<br>S-ACTIVITY-INTERRUPT response<br>S-ACTIVITY-INTERRUPT confirm                                    | ACTIVITY INTERRUPT SPDU<br>ACTIVITY INTERRUPT SPDU<br>ACTIVITY INTERRUPT ACK SPDU<br>ACTIVITY INTERRUPT ACK SPDU |
| Activity discard   | S-ACTIVITY-DISCARD request<br>S-ACTIVITY-DISCARD indication<br>S-ACTIVITY-DISCARD response<br>S-ACTIVITY-DISCARD confirm                                            | ACTIVITY DISCARD SPDU<br>ACTIVITY DISCARD SPDU<br>ACTIVITY DISCARD ACK SPDU<br>ACTIVITY DISCARD ACK SPDU         |
| Activity end       | S-ACTIVITY-END request<br>S-ACTIVITY-END indication<br>S-ACTIVITY-END response<br>S-ACTIVITY-END confirm                                                            | ACTIVITY END SPDU<br>ACTIVITY END SPDU<br>ACTIVITY END ACK SPDU<br>ACTIVITY END ACK SPDU                         |
| Orderly release    | S-RELEASE request<br>S-RELEASE indication<br>S-RELEASE (accept) response<br>S-RELEASE (accept) confirm<br>S-RELEASE (reject) response<br>S-RELEASE (reject) confirm | FINISH SPDU<br>FINISH SPDU<br>DISCONNECT SPDU<br>DISCONNECT SPDU<br>NOT FINISHED SPDU<br>NOT FINISHED SPDU       |
| U-Abort            | S-U-ABORT request<br>S-U-ABORT indication                                                                                                                           | ABORT SPDU<br>ABORT SPDU                                                                                         |
| P-Abort            | S-P-ABORT indication                                                                                                                                                | ABORT SPDU                                                                                                       |

Table 2 – Transport service primitives

| Primitives                             | X/Y | Parameters                                                                                       |
|----------------------------------------|-----|--------------------------------------------------------------------------------------------------|
| T-CONNECT request<br>indication        | X   | Called address<br>Calling address<br>Expedited data option<br>Quality of service<br>TS-user-data |
| T-CONNECT response<br>confirm          | X   | Quality of service<br>Responding address<br>Expedited data option<br>TS-user-data                |
| T-DATA request<br>indication           | X   | TS-user-data                                                                                     |
| T-EXPEDITED-DATA request<br>indication | Y   | TS-user-data                                                                                     |
| T-DISCONNECT request                   | X   | TS-user-data                                                                                     |
| T-DISCONNECT indication                | X   | Disconnect reason<br>TS-user-data                                                                |

Key :

X : The session protocol assumes that this service is always available.

Y : The session protocol assumes that this service is provided by the Transport Layer when requested by the SPM during the session connection establishment phase.

## 5.4 Functions of the Session Layer

### 5.4.1 Overview of functions

The functions in the Session Layer are those necessary to bridge the gap between the services available from the Transport Layer and those offered to the SS-users.

The functions in the Session Layer are concerned with dialogue management, data flow synchronization, and data flow resynchronization.

These functions are described below; the descriptions are grouped into those concerned with the connection establishment phase, the data transfer phase, and the release phase.

### 5.4.2 Connection establishment phase

The purpose of the connection establishment phase is to establish a session connection between two SS-users, and

- a) to map session addresses onto transport addresses;
- b) to select transport quality of service parameters needed (see 6.1.4);
- c) to negotiate session parameters (see 7.1 and 7.2);
- d) to transfer session selectors (see 7.1 and 7.2) if required;
- e) to distinguish between session connections (see 7.1 and 7.2);
- f) to transfer a limited amount of transparent user data (see 7.1 and 7.2).

### 5.4.3 Data transfer phase

The purpose of the data transfer phase is to transport SSDUs between two SS-users connected by a session connection. This purpose is achieved by means of transmission of SPDUs and by the following functions, each of which may or may not be used, depending on the functional units selected in the session connection establishment phase. These concepts are defined in ISO 8326 :

- a) normal data transfer (see 7.9), which may involve segmenting of SSDUs into SPDUs and reassembly by the destination SPM; and concatenation and separation of certain SPDUs. There are two modes of operation :
  - 1) half-duplex, when the right to send data is restricted to the owner of the data token;
  - 2) duplex, when there is no restriction on the right to send data.
- b) token management (see 7.14 to 7.17), to enable the SS-users to request and transfer tokens which control the exclusive right to exercise certain functions (see table 5);

c) exception reporting (see 7.25 and 7.26), to enable the SS-provider or the SS-user to report exception conditions that are less than those requiring abort;

d) typed data transfer (see 7.11), to enable transfer of information which is not subject to assignment of the data token;

e) minor synchronization point (see 7.18 and 7.19), to enable the SS-users to define minor synchronization points in the normal data flow. These minor synchronization points may optionally be confirmed, but have no implications on the data flow. Minor synchronization points are identified by synchronization point serial numbers. The serial number is incremented by one on each occasion that a minor synchronization point is placed in the data flow, and each time a minor synchronization point is received, such that both SS-users have the same serial numbers for the same synchronization point;

f) major synchronization point (see 7.20 and 7.21 and e) above), to enable the SS-users to define major synchronization points in the normal data flow. These major synchronization points are required to be confirmed before the requesting SS-user is permitted to send any subsequent data on either the normal flow or the expedited flow and as such clearly separate the dialogue units;

g) resynchronize (see 7.22 and 7.23), a function that allows a session connection to be set or reset to a defined synchronization point and reassign the tokens;

h) expedited data transfer (see 7.10), a function used to convey a limited amount of user data with special handling. Such data may bypass normal data en route, but will be delivered prior to any data subsequently sent on the transport normal flow or the transport expedited flow;

i) activity management (see 7.27 to 7.34) provides a means explicitly to start, end, assume, interrupt or discard an activity. This provides a way

- 1) to identify the entered activity and commence synchronization point serial numbering;
- 2) to identify the continued activity and reset the synchronization point serial number in case of resumption;

j) capability data exchange (see 7.12 and 7.13), to provide a confirmed transfer of a limited amount of user data.

### 5.4.4 Connection release phase

The purpose of the release phase is to provide disconnection of the session connection, by using the following functions :

- a) orderly release (negotiated and non-negotiated);
- b) abort (provider and user initiated);
- c) transfer of a limited amount of transparent user data.

## 5.5 Functional units

Functional units are logical groupings of related elements of procedure defined by this International Standard for the purpose of

- a) negotiation for use during session connection establishment;
- b) specification of conformance requirements.

The SPDUs associated with elements of procedure for each functional unit are specified in table 3.

Tokens are associated with functional units (see 5.6).

### 5.5.1 Kernel functional unit

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

### 5.5.2 Negotiated release functional unit

The negotiated release functional unit supports the negotiated release service which enables the SS-users to negotiate the orderly release of the session connection. If this functional unit has been selected, an attempt to release the session connection may be refused by the accepting SS-user.

### 5.5.3 Half-duplex functional unit

The half-duplex functional unit is used to control the right to send data. It is not valid to select both this functional unit and the duplex functional unit for use on the same session connection.

### 5.5.4 Duplex functional unit

The duplex functional unit is used when the right to send data is not controlled. It is not valid to select both this functional unit and the half-duplex functional unit for use on the same session connection.

### 5.5.5 Expedited data functional unit

The expedited data functional unit supports the expedited data service and allows the transfer of a limited amount of SS-user data.

The services supported by this functional unit can only be requested when the transport expedited flow is available to this session connection.

### 5.5.6 Typed data functional unit

The typed data functional unit enables the SS-users to transfer data in a manner which is not subject to the control imposed by the availability of the data token.

### 5.5.7 Capability data exchange functional unit

The capability data functional unit supports the capability data exchange service, which allows a confirmed transfer of a limited amount of SS-user data when the activity management functional unit has been selected, but when no activity is in progress.

### 5.5.8 Minor synchronize functional unit

The minor synchronize functional unit supports the minor synchronization service which enables the SS-user to request that the SPM places minor synchronization points in the normal data flow. These minor synchronization points are identified by serial numbers.

### 5.5.9 Major synchronize functional unit

The major synchronize functional unit supports the major synchronize service which enables the SS-user to request that the SPM places major synchronization points in the normal data flow. These major synchronization points are identified by serial numbers, and clearly separate the data flow before and after the major synchronization point.

### 5.5.10 Resynchronize functional unit

The resynchronize functional unit supports the resynchronize service which enables the SS-users to modify the synchronization point serial number and reassign the tokens.

### 5.5.11 Exceptions functional unit

The exceptions functional unit allows both the SPM and the SS-users to report detected errors, rather than aborting the session connection.

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

### 5.5.12 Activity management functional unit

The activity management functional unit supports the activity management services which allows the SS-users to manage synchronized logical pieces of work.

## 5.6 Tokens

Table 4 specifies those functional units that have tokens associated with them.

The SPM may only send an SPDU listed in table 5 (and accept the associated service primitive) subject to the availability and assignment of tokens defined in that table.

Table 3 — Functional units

| Functional unit          | SPDU code | SPDU name                        | Reference |
|--------------------------|-----------|----------------------------------|-----------|
| Kernel                   | CN        | CONNECT (see note 1)             | 7.1       |
|                          | AC        | ACCEPT (see note 1)              | 7.2       |
|                          | RF        | REFUSE (see note 1)              | 7.3       |
|                          | FN        | FINISH                           | 7.4       |
|                          | DN        | DISCONNECT                       | 7.5       |
|                          | AB        | ABORT                            | 7.7       |
|                          | AA        | ABORT ACCEPT (see note 2)        | 7.8       |
|                          | DT        | DATA TRANSFER                    | 7.9       |
| Negotiated release       | NF        | NOT FINISHED                     | 7.6       |
|                          | GT        | GIVE TOKENS (see note 4)         | 7.14      |
|                          | PT        | PLEASE TOKENS (see note 4)       | 7.15      |
| Half-duplex              | GT        | GIVE TOKENS (see note 3)         | 7.14      |
|                          | PT        | PLEASE TOKENS (see note 3)       | 7.15      |
| Duplex                   |           | No additional associated SPDUs   |           |
| Expedited data           | EX        | EXPEDITED DATA                   | 7.10      |
| Typed data               | TD        | TYPED DATA                       | 7.11      |
| Capability data exchange | CD        | CAPABILITY DATA                  | 7.12      |
|                          | CDA       | CAPABILITY DATA ACK              | 7.13      |
| Minor synchronize        | MIP       | MINOR SYNC POINT                 | 7.18      |
|                          | MIA       | MINOR SYNC ACK                   | 7.19      |
|                          | GT        | GIVE TOKENS (see note 5)         | 7.14      |
|                          | PT        | PLEASE TOKENS (see note 5)       | 7.15      |
| Major synchronize        | MAP       | MAJOR SYNC POINT                 | 7.20      |
|                          | MAA       | MAJOR SYNC ACK                   | 7.21      |
|                          | PR        | PREPARE (see note 6)             | 7.24      |
|                          | GT        | GIVE TOKENS (see note 7)         | 7.14      |
|                          | PT        | PLEASE TOKENS (see note 7)       | 7.15      |
| Resynchronize            | RS        | RESYNCHRONIZE                    | 7.22      |
|                          | RA        | RESYNCHRONIZE ACK                | 7.23      |
|                          | PR        | PREPARE (see note 6)             | 7.24      |
| Exceptions               | ER        | EXCEPTION REPORT                 | 7.25      |
|                          | ED        | EXCEPTION DATA                   | 7.26      |
| Activity management      | AS        | ACTIVITY START                   | 7.27      |
|                          | AR        | ACTIVITY RESUME                  | 7.28      |
|                          | AI        | ACTIVITY INTERRUPT               | 7.29      |
|                          | AIA       | ACTIVITY INTERRUPT ACK           | 7.30      |
|                          | AD        | ACTIVITY DISCARD                 | 7.31      |
|                          | ADA       | ACTIVITY DISCARD ACK             | 7.32      |
|                          | AE        | ACTIVITY END                     | 7.33      |
|                          | AEA       | ACTIVITY END ACK                 | 7.34      |
|                          | PR        | PREPARE (see note 6)             | 7.24      |
|                          | GT        | GIVE TOKENS (see note 7)         | 7.14      |
|                          | PT        | PLEASE TOKENS (see note 7)       | 7.15      |
|                          | GTC       | GIVE TOKENS CONFIRM (see note 8) | 7.16      |
|                          | GTA       | GIVE TOKENS ACK (see note 8)     | 7.17      |

## NOTES

- 1 An implementation (see clause 9) is required to be able to
  - a) send a CONNECT SPDU and receive an ACCEPT SPDU or a REFUSE SPDU, or
  - b) receive a CONNECT SPDU and send an ACCEPT SPDU or a REFUSE SPDU, or
  - c) send and receive both.
- 2 Reception and correct action is mandatory; transmission is optional if the transport connection is not to be reused (see 7.8.2).
- 3 Used to manage the data token.
- 4 Used to manage the release token.
- 5 Used to manage the synchronize-minor token.
- 6 PREPARE SPDU is mandatory if the transport expedited flow is available to this session connection, otherwise it is not used (see 6.4).
- 7 Used to manage the major/activity token.
- 8 Used only on session connections on which activity management has been selected, for giving all available tokens, when no activity is in progress.

**Table 4 — Tokens associated with functional units**

| 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 5 — Token restrictions**

| SPDUs                            | data token | synchronize-minor token | major/activity token | release token |
|----------------------------------|------------|-------------------------|----------------------|---------------|
| FINISH SPDU                      | 2          | 2                       | 2                    | 2             |
| NOT FINISHED SPDU                | nr         | nr                      | nr                   | 0             |
| DATA TRANSFER SPDU (half-duplex) | 1          | nr                      | nr                   | nr            |
| DATA TRANSFER SPDU (duplex)      | 3          | nr                      | nr                   | nr            |
| CAPABILITY DATA SPDU             | 2          | 2                       | 1                    | nr            |
| GIVE TOKEN SPDU                  | 1          | nr                      | nr                   | nr            |
| data token                       | nr         | 1                       | nr                   | nr            |
| synchronize-minor token          | nr         | nr                      | 1                    | nr            |
| major/activity token             | nr         | nr                      | nr                   | 1             |
| release token                    |            |                         |                      |               |
| PLEASE TOKEN SPDU                | 0          | nr                      | nr                   | nr            |
| data token                       | nr         | 0                       | nr                   | nr            |
| synchronize-minor token          | nr         | nr                      | 0                    | nr            |
| major/activity token             | nr         | nr                      | nr                   | 0             |
| release token                    |            |                         |                      |               |
| GIVE TOKENS CONFIRM SPDU         | 2          | 2                       | 1                    | 2             |
| MINOR SYNC POINT SPDU            | 2          | 1                       | nr                   | nr            |
| MAJOR SYNC POINT SPDU            | 2          | 2                       | 1                    | nr            |
| EXCEPTION REPORT SPDU            | 0          | nr                      | nr                   | nr            |
| EXCEPTION DATA SPDU              | 0          | nr                      | nr                   | nr            |
| ACTIVITY START SPDU              | 2          | 2                       | 1                    | nr            |
| ACTIVITY RESUME SPDU             | 2          | 2                       | 1                    | nr            |
| ACTIVITY INTERRUPT SPDU          | nr         | nr                      | 1                    | nr            |
| ACTIVITY DISCARD SPDU            | nr         | nr                      | 1                    | nr            |
| ACTIVITY END SPDU                | 2          | 2                       | 1                    | nr            |

Key :

0 : Token available and not assigned to the SS-user who initiated the associated service primitive.

1 : Token available and assigned to the SS-user who initiated the associated service primitive.

2 : Token not available or token assigned to the SS-user who initiated the associated service primitive.

3 : Token not available.

nr : No restriction.

## 5.7 Negotiation

Negotiation takes place between both SPMs during session connection establishment according to the following rules.

### 5.7.1 Negotiation of functional units

Each SPM proposes use or non-use of each functional unit, except for the kernel functional unit, based on requirements from the SS-users. The functional unit is selected only if both the initiator and the responder propose use of the functional unit.

The capability data exchange functional unit can only be proposed if the activity management functional unit is also proposed.

The exceptions functional unit can only be proposed if the half-duplex functional unit is also proposed.

### 5.7.2 Negotiation of initial token settings

When the initiator proposes use of a functional unit that requires a token, it also proposes the initial token setting :

- a) initiator's side;
- b) responder's side;
- c) called SS-user's choice.

If use of the functional unit is selected, the token is set to the side proposed by the initiator. If the initiator proposed "called SS-user choice", the responder's proposed token setting is selected.

### 5.7.3 Negotiation of initial serial number

When the initiator proposes any of the minor synchronize, major synchronize or resynchronize functional units but does not propose the activity management functional unit, it also proposes an initial serial number.

When the initiator proposes any of the minor synchronize, major synchronize or resynchronize functional units and also proposes the activity management functional unit, it may also propose an initial serial number.

In all other cases, the initiator does not propose an initial serial number.

When the responder proposes any of the minor synchronize, major synchronize or resynchronize functional units but does not propose the activity management functional unit, it also proposes an initial serial number, which is the first serial number to be used.

In all other cases, the responder does not propose an initial serial number.

### 5.7.4 Negotiation of version number

Each SPM indicates all versions of the protocol that it is capable of supporting. The highest common version number is used.

### 5.7.5 Negotiation of maximum TSDU size

Each SPM proposes a maximum TSDU size that the initiator is permitted to send in the data transfer and connection release phases. The lesser of the two numbers is used. A zero value is interpreted to mean unlimited TSDU size. If either SPM proposes zero, the initiator may not send segmented SSDUs on the session connection.

Each SPM also proposes a maximum TSDU size that the responder is permitted to send in the data transfer and connection release phases. The lesser of the two numbers is used. A zero value is interpreted to mean unlimited TSDU size. If either SPM proposes zero, the responder may not send segmented SSDUs on the session connection.

## 5.8 Local variables

This International Standard uses local variables as a means of clarifying the effect of certain actions and clarifying the conditions under which certain actions are valid.

### 5.8.1 Vact

Vact is used by the SPM to determine if an activity is in progress when the activity management functional unit has been selected :

Vact = true : an activity is in progress;

Vact = false : no activity is in progress.

### 5.8.2 Vnextact

Vnextact is used by the SPM when the activity management functional unit has been selected :

Vnextact = true : a MAJOR SYNC POINT SPDU has been sent or received;

Vnextact = false : an ACTIVITY END SPDU has been sent or received.

### 5.8.3 V(A)

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

### 5.8.4 V(M)

V(M) is used by the SPM and is the next serial number to be used.

### 5.8.5 V(R)

V(R) is used by the SPM and is the lowest serial number to which resynchronization restart is permitted.

### 5.8.6 Vsc

Vsc is used by the SPM to determine whether or not the SS-user has the right to send minor synchronization point responses. Vsc has the following values :

**Vsc = true** : the SS-user has the right to issue minor synchronization point responses when V(A) is less than V(M);

**Vsc = false** : the SS-user does not have the right to issue minor synchronization point responses.

**NOTE** — The manipulation of V(A), V(M), V(R) and Vsc and the circumstances under which they are updated are specified in clause 7 and are summarized in table 42 in annex A.

## Section two : Session protocol specification

### 6 Use of the transport service

This clause defines the way that the transport service primitives are used by the SPM.

#### 6.1 Assignment of a session connection to the transport connection

##### 6.1.1 Purpose

Assignment of a session connection to a transport connection.

##### 6.1.2 Transport service primitives

The procedure uses the following transport service primitives :

- T-CONNECT request
- T-CONNECT indication
- T-CONNECT response
- T-CONNECT confirm
- T-DISCONNECT request
- T-DISCONNECT indication

##### 6.1.3 SPDUs used

No SPDUs are used during assignment to a transport connection.

##### 6.1.4 Description

A session connection is assigned to an existing transport connection suitable for reuse, or a new transport connection is created for the purpose. This assignment is based on the quality of service (see ISO 8326) requested by the SS-user in the S-CONNECT request.

If a transport connection is established with the transport expedited data option, the transport expedited flow is available for the duration of the transport connection. Use of transport expedited data is specified in 6.4.

The transport expedited flow is requested by the SPM when the T-CONNECT request is issued if

- a) the SS-user requested the expedited data functional unit; or
- b) the SS-user requested an extended control QOS for the session connection.

Only the initiator of the transport connection is permitted to issue the CONNECT SPDU.

When a session connection is terminated, the underlying transport connection is also terminated, unless reuse of the transport connection has been agreed.

Use of the TS-user data parameter in T-CONNECT request, indication, response and confirm is reserved for future use. When a T-CONNECT request or a T-CONNECT response is

issued, this parameter is empty. When a T-CONNECT indication or T-CONNECT confirm is received, this parameter is ignored.

#### 6.2 Reuse of the transport connection

##### 6.2.1 Purpose

To allow the transport connection to be retained for reuse by another session connection.

##### 6.2.2 Transport service primitives

The procedure uses the following transport service primitives :

- T-DATA request
- T-DATA indication

##### 6.2.3 SPDUs used

The following SPDUs are related to reuse of the transport connection :

- REFUSE SPDU (see 7.3)
- FINISH SPDU (see 7.4)
- DISCONNECT SPDU (see 7.5)
- ABORT SPDU (see 7.7)
- ABORT ACCEPT SPDU (see 7.8)

##### 6.2.4 Description

When a session connection is refused, or has been successfully connected and subsequently disconnected, by abort or orderly release, the supporting transport connection may be either disconnected or reused.

The transport connection may be kept for reuse provided that the transport expedited flow is not available, and either

- a) the SPM which established the transport connection requests retention of the transport connection by parameter in an ABORT SPDU or a FINISH SPDU, or
- b) the SPM which established the transport connection receives a REFUSE SPDU or an ABORT SPDU which indicates by parameter that the transport connection is to be retained.

To avoid contention for a retained transport connection, only the transport connection initiator may reuse the transport connection by sending a CONNECT SPDU to establish a new session connection.

### 6.3 Use of transport normal data

#### 6.3.1 Purpose

To convey SPDUs in user data fields of transport service normal data primitives.

### 6.3.2 Transport service primitives

The procedure uses the following transport service primitives :

T-DATA request  
T-DATA indication

### 6.3.3 SPDUs used

The following SPDUs are sent on the transport normal flow :

CONNECT SPDU (see 7.1)  
ACCEPT SPDU (see 7.2)  
REFUSE SPDU (see 7.3)  
FINISH SPDU (see 7.4)  
DISCONNECT SPDU (see 7.5)  
NOT FINISHED SPDU (see 7.6)  
DATA TRANSFER SPDU (see 7.9)  
TYPED DATA SPDU (see 7.11)  
CAPABILITY DATA SPDU (see 7.12)  
CAPABILITY DATA ACK SPDU (see 7.13)  
GIVE TOKENS SPDU (see 7.14)  
PLEASE TOKENS SPDU (see 7.15)  
GIVE TOKENS CONFIRM SPDU (see 7.16)  
GIVE TOKENS ACK SPDU (see 7.17)  
MINOR SYNC POINT SPDU (see 7.18)  
MINOR SYNC ACK SPDU (see 7.19)  
MAJOR SYNC POINT SPDU (see 7.20)  
MAJOR SYNC ACK SPDU (see 7.21)  
RESYNCHRONIZE SPDU (see 7.22)  
RESYNCHRONIZE ACK SPDU (see 7.23)  
EXCEPTION REPORT SPDU (see 7.25)  
EXCEPTION DATA SPDU (see 7.26)  
ACTIVITY START SPDU (see 7.27)  
ACTIVITY RESUME SPDU (see 7.28)  
ACTIVITY INTERRUPT SPDU (see 7.29)  
ACTIVITY INTERRUPT ACK SPDU (see 7.30)  
ACTIVITY DISCARD SPDU (see 7.31)  
ACTIVITY DISCARD ACK SPDU (see 7.32)  
ACTIVITY END SPDU (see 7.33)  
ACTIVITY END ACK SPDU (see 7.34)

If the transport expedited flow is not available, the following additional SPDUs are sent on the transport normal flow :

ABORT SPDU (see 7.7)  
ABORT ACCEPT SPDU (see 7.8)

### 6.3.4 Transfer of SPDUs

The SPDUs listed in 6.3.3 are transferred using the transport normal data transfer service.

### 6.3.5 Segmenting

Each SSDU is mapped one-to-one onto an SPDU, unless segmenting has been selected for that direction of transfer, in which case a data SSDU or a typed SSDU may be mapped onto more than one SPDU.

NOTE — Implementors should note that when segmenting is selected

- a) the control information of each SPDU indicates whether or not it contains the first or last segment of the SSDU;

- b) the size of the segments of the SSDU is constrained by the maximum TSDU size selected for that direction of transfer.

### 6.3.6 Maximum TSDU size

When a maximum TSDU size has been selected, the SPDU size may not exceed the maximum TSDU size selected for that direction of transfer and a sequence of concatenated SPDUs may not exceed the maximum TSDU size selected for that direction of transfer.

### 6.3.7 Concatenation

Each SPDU is defined in table 6 as belonging to one of the following categories :

- a) category 0 SPDUs which may be mapped one-to-one onto a TSDU or may be concatenated with one or more category 2 SPDUs;
- b) category 1 SPDUs which are always mapped one-to-one onto a TSDU;
- c) category 2 SPDUs which are never mapped one-to-one onto a TSDU.

Basic concatenations of a category 0 SPDU with a single category 2 SPDU, defined as valid and in the order indicated in table 7, may always be mapped onto a single TSDU.

If the receiving SPM has indicated that it can accept extended concatenation, the sending SPM may map a category 0 SPDU with one or more category 2 SPDUs (as specified in table 8) onto a single TSDU.

The valid mappings of SPDUs onto TSDUs are illustrated in figure 3.

Any other concatenation of SPDUs is defined as invalid.

#### 6.3.7.1 Processing order of concatenated SPDUs

On receipt of SPDUs that have been concatenated using basic concatenation, the category 2 SPDUs are processed before the category 0 SPDU.

On receipt, SPDUs that have been concatenated using extended concatenation are processed in the following order :

- a) ACTIVITY START SPDU or  
ACTIVITY RESUME SPDU;
- b) DATA TRANSFER SPDU;
- c) MINOR SYNC POINT SPDU or  
MINOR SYNC ACK SPDU or  
MAJOR SYNC POINT SPDU or  
MAJOR SYNC ACK SPDU or  
ACTIVITY END SPDU or  
ACTIVITY END ACK SPDU;
- d) GIVE TOKENS SPDU or  
PLEASE TOKENS SPDU.

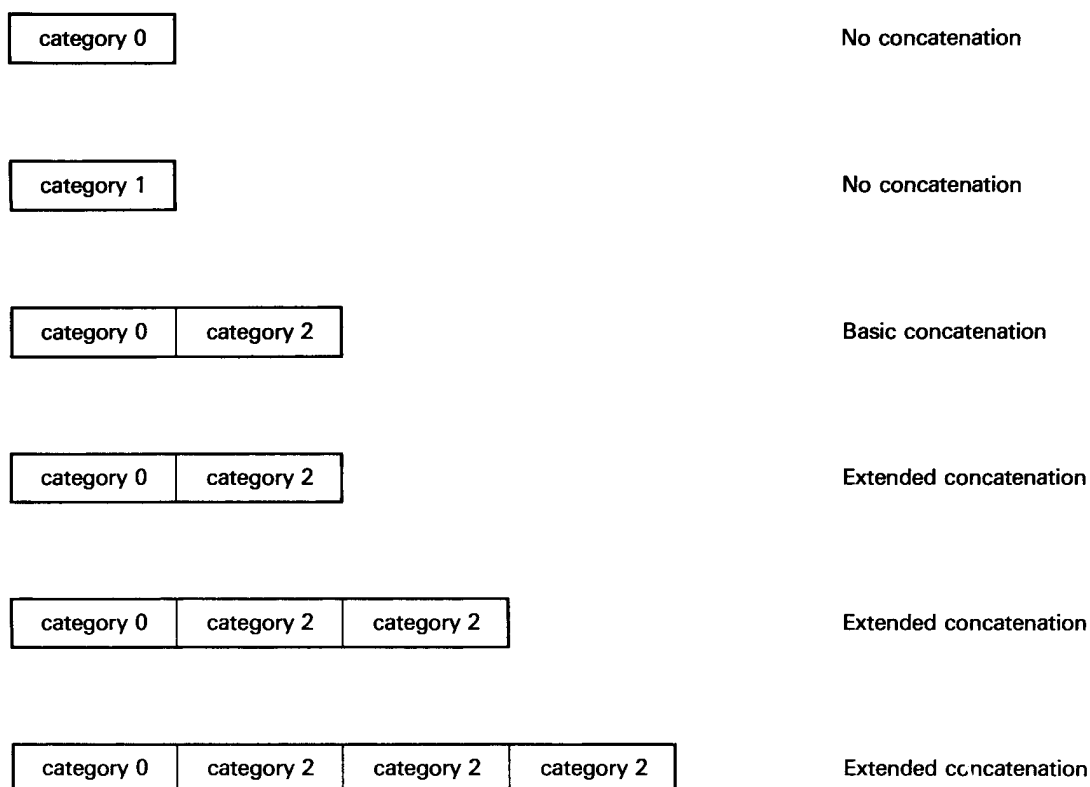


**Figure 3 – Illustration of TSDU structures**

**Table 6 — Category 0, 1 and 2 SPDUs**

| Category 0 SPDUs                       | Category 1 SPDUs                                                                                                                                                                                                                              | Category 2 SPDUs                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| GIVE TOKENS SPDU<br>PLEASE TOKENS SPDU | CONNECT SPDU<br>ACCEPT SPDU<br>REFUSE SPDU<br>FINISH SPDU<br>DISCONNECT SPDU<br>NOT FINISHED SPDU<br>ABORT SPDU<br>ABORT ACCEPT SPDU<br>GIVE TOKENS CONFIRM SPDU<br>GIVE TOKENS ACK SPDU<br>EXPEDITED SPDU<br>PREPARE SPDU<br>TYPED DATA SPDU | DATA TRANSFER SPDU<br>MINOR SYNC POINT SPDU<br>MINOR SYNC ACK SPDU<br>MAJOR SYNC POINT SPDU<br>MAJOR SYNC ACK SPDU<br>RESYNCHRONIZE SPDU<br>RESYNCHRONIZE ACK SPDU<br>ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU<br>ACTIVITY DISCARD SPDU<br>ACTIVITY DISCARD ACK SPDU<br>ACTIVITY INTERRUPT SPDU<br>ACTIVITY INTERRUPT ACK SPDU<br>ACTIVITY END SPDU<br>ACTIVITY END ACK SPDU<br>CAPABILITY DATA SPDU<br>CAPABILITY DATA ACK SPDU<br>EXCEPTION REPORT SPDU<br>EXCEPTION DATA SPDU |

**Table 7 — Valid basic concatenation of SPDUs**

| First SPDU                               | Second SPDU                                            |
|------------------------------------------|--------------------------------------------------------|
| GIVE TOKENS SPDU**                       | DATA TRANSFER SPDU                                     |
| GIVE TOKENS SPDU<br>PLEASE TOKENS SPDU   | MINOR SYNC POINT SPDU<br>MINOR SYNC ACK SPDU           |
| GIVE TOKENS SPDU<br>PLEASE TOKENS SPDU   | MAJOR SYNC POINT SPDU<br>MAJOR SYNC ACK SPDU           |
| GIVE TOKENS SPDU*<br>PLEASE TOKENS SPDU  | RESYNCHRONIZE SPDU<br>RESYNCHRONIZE ACK SPDU           |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU     | ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU            |
| GIVE TOKENS SPDU*<br>PLEASE TOKENS SPDU  | ACTIVITY DISCARD SPDU<br>ACTIVITY DISCARD ACK SPDU     |
| GIVE TOKENS SPDU*<br>PLEASE TOKENS SPDU  | ACTIVITY INTERRUPT SPDU<br>ACTIVITY INTERRUPT ACK SPDU |
| GIVE TOKENS SPDU<br>PLEASE TOKENS SPDU   | ACTIVITY END SPDU<br>ACTIVITY END ACK SPDU             |
| GIVE TOKENS SPDU*<br>PLEASE TOKENS SPDU  | CAPABILITY DATA SPDU<br>CAPABILITY DATA ACK SPDU       |
| PLEASE TOKENS SPDU<br>PLEASE TOKENS SPDU | EXCEPTION REPORT SPDU<br>EXCEPTION DATA SPDU           |

Key :

- \* Indicates that the Token Item parameter is not present in the GIVE TOKENS SPDU. In all other cases, the Token Item parameter may or may not be present.
- \*\* Indicates that the Token Item parameter is only present in the GIVE TOKENS SPDU if this DATA TRANSFER SPDU contains a complete SSDU, or the last segment of a segmented SSDU.

Table 8 — Valid extended concatenation of SPDUs

| First SPDU                                                                                                               | Second SPDU                                                                                                                               | Third SPDU                                                                                                                                 | Fourth SPDU                                                                                                                      | Status                     |
|--------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|----------------------------|
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU                                                                                     | MINOR SYNC ACK SPDU<br>MAJOR SYNC ACK SPDU                                                                                                |                                                                                                                                            |                                                                                                                                  |                            |
| GIVE TOKENS SPDU                                                                                                         | ACTIVITY END ACK SPDU                                                                                                                     |                                                                                                                                            |                                                                                                                                  |                            |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU* | ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU<br>ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU<br>ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU | MINOR SYNC POINT SPDU<br>MINOR SYNC POINT SPDU<br>ACTIVITY END SPDU<br>ACTIVITY END SPDU<br>MAJOR SYNC POINT SPDU<br>MAJOR SYNC POINT SPDU |                                                                                                                                  |                            |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU<br>GIVE TOKENS SPDU<br>GIVE TOKENS SPDU                                             | MINOR SYNC POINT SPDU<br>MINOR SYNC ACK SPDU<br>MAJOR SYNC POINT SPDU<br>MAJOR SYNC ACK SPDU                                              | DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU                                                       |                                                                                                                                  | CL<br>CL<br>CL<br>CL       |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU                                                                                     | ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU                                                                                               | DATA TRANSFER SPDU<br>DATA TRANSFER SPDU                                                                                                   |                                                                                                                                  | CF<br>CF                   |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU                                                                                     | ACTIVITY END SPDU<br>ACTIVITY END ACK SPDU                                                                                                | DATA TRANSFER SPDU<br>DATA TRANSFER SPDU                                                                                                   |                                                                                                                                  | CL<br>CL                   |
| GIVE TOKENS SPDU<br>GIVE TOKENS SPDU<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU*<br>GIVE TOKENS SPDU* | ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU<br>ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU<br>ACTIVITY START SPDU<br>ACTIVITY RESUME SPDU | MINOR SYNC POINT SPDU<br>MINOR SYNC POINT SPDU<br>ACTIVITY END SPDU<br>ACTIVITY END SPDU<br>MAJOR SYNC POINT SPDU<br>MAJOR SYNC POINT SPDU | DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU<br>DATA TRANSFER SPDU | C<br>C<br>C<br>C<br>C<br>C |

Key :

\* indicates that the Token Item parameter is not present in the GIVE TOKENS SPDU.

Status :

CL : The DATA TRANSFER SPDU contains a complete SPDU or the last segment of an SSDU.

CF : The DATA TRANSFER SPDU contains a complete SSDU or the first segment of an SSDU. In the latter case, the Token Item parameter is not present in the GIVE TOKENS SPDU.

C : The DATA TRANSFER SPDU contains a complete SSDU.

## 6.4 Use of transport expedited data

### 6.4.1 Purpose

To convey SPDUs on a separate transport flow.

### 6.4.2 Transport service primitives

The procedure uses the following transport service primitives :

- T-EXPEDITED-DATA request
- T-EXPEDITED-DATA indication

### 6.4.3 SPDUs used

The following SPDUs are sent on the transport expedited flow when it is available :

- ABORT SPDU (see 7.7)
- ABORT ACCEPT SPDU (see 7.8)
- EXPEDITED DATA SPDU (see 7.10)
- PREPARE SPDU (see 7.24)

### 6.4.4 Description

The SPDUs listed in 6.4.3 are sent on the transport expedited flow if it is selected, and may be used to bypass any flow control restrictions or congestion on the transport normal flow. SPDUs sent on the transport expedited flow may be delivered to the accepting SS-user earlier than SSDUs submitted previously by the sending SS-user and sent on the transport normal flow, but no later than subsequently submitted SSDUs.

When the transport expedited flow is not available

- a) EXPEDITED DATA SPDUs are not sent;
- b) ABORT and ABORT ACCEPT SPDUs are sent on the transport normal flow;
- c) PREPARE SPDUs are not sent.

The use of the Disconnect Reason parameter in T-DISCONNECT indication is a local matter.

## 6.5 Flow control

There is no peer flow control in the session layer. To prevent the SS-users from being overloaded with data, the receiving SPM may apply back pressure across the transport connection, using the transport flow control. The decision on when or how back pressure is applied is a local matter.

## 6.6 Transport disconnection

### 6.6.1 Purpose

To release a transport connection.

### 6.6.2 Transport service primitives

The procedure uses the following transport service primitives :

- T-DISCONNECT request
- T-DISCONNECT indication

### 6.6.3 SPDUs used

No SPDUs are used.

### 6.6.4 Description

After the session connection has been released or aborted and the transport connection is not to be reused, the transport connection is disconnected.

When a T-DISCONNECT indication is received, as a result of an error detected by the transport service provider, the SPM issues an S-P-ABORT indication to the local SS-user.

When issuing a T-DISCONNECT request, the SPM may optionally use the T-DISCONNECT user data field to indicate the reason for the transport disconnection to the remote SPM. The reason code consists of one octet with the following values :

- a) 0 — session protocol error for which an ABORT SPDU could not be sent;
- b) 1 — normal transport disconnection when the transport connection is not to be reused;
- c) 2 — normal transport disconnection when the transport connection was to be reused, but reuse is not possible for local reasons.

## 7 Elements of procedure related to SPDUs

This clause defines valid sequences of operation of the protocol.

A more precise definition of procedures is contained in annex A which incorporates all the checks to determine the validity of a particular event at a particular point in time. In case of arbitration or dispute, annex A takes precedence over this clause.

### 7.1 CONNECT SPDU

The CONNECT SPDU is transmitted by the initiator of the transport connection on a previously assigned transport connection in order to initiate a session connection.

#### 7.1.1 Content of CONNECT SPDU

The CONNECT SPDU contains

- a) a Connection Identifier parameter group, which is supplied by the calling SS-user, to enable the SS-users to identify this specific session connection. This parameter group has no effect on the SPM. It contains
  - 1) a Calling SS-user Reference parameter;
  - 2) a Common Reference parameter;
  - 3) an Additional Reference Information parameter.

- b) a Connect/Accept Item parameter group containing
  - 1) a Protocol Options parameter which enables the initiator to indicate its ability to receive extended concatenated SPDUs;
  - 2) a TSDU maximum size parameter which, if present and not zero, indicates that segmenting of SSDUs is requested by the initiator. The initiator proposes the maximum TSDU sizes for each direction of transfer (see 5.7.5). If segmenting is not to be used, this parameter is not present or is zero;
  - 3) a Version Number parameter to identify which versions of this protocol have been implemented;
  - 4) an Initial Serial Number parameter which is proposed by the calling SS-user in the case where the activity management functional unit is not proposed and any of the minor synchronize, major synchronize or resynchronize functional units are proposed. As an SS-user option, an Initial Serial Number parameter may be proposed even if the activity management functional unit is proposed provided that any of the minor synchronize, major synchronize or resynchronize functional units are also proposed;
  - 5) a Token Setting Item parameter supplied by the calling SS-user, which proposes the initial token positions for each token available on this connection, as derived from the functional units proposed in the Session User Requirements parameter (see table 4). The initial token positions can be specified to be on the initiator's side or on the acceptor's side or the initiator can specify that the decision is to be made by the called SS-user.
- c) a Session User Requirements parameter containing a list of the functional units proposed by the calling SS-user. At least one of the half-duplex and the duplex functional units shall be proposed. The SPM is required to provide the associated protocol functions.
- d) Calling Session Selector and Called Session Selector parameters corresponding to the calling SS-user and the called SS-user may be present and are derived from session addresses provided by the calling SS-user.
- e) a User Data parameter which allows a limited amount of transparent user data to be passed from the calling SS-user to the called SS-user.

### 7.1.2 Sending the CONNECT SPDU

An S-CONNECT request results in the assignment of a transport connection. When the transport connection is established, a CONNECT SPDU is sent on the transport normal flow. The SPM waits until it receives an ACCEPT SPDU or a REFUSE SPDU.

### 7.1.3 Receiving the CONNECT SPDU

A valid incoming CONNECT SPDU results in an S-CONNECT indication to an SS-user, according to the Called Session Selector parameter of the CONNECT SPDU. The SPM then waits for an S-CONNECT response from the called SS-user.

## 7.2 ACCEPT SPDU

An SPM receiving a CONNECT SPDU may accept a proposal to establish a session connection by transferring an ACCEPT SPDU to the initiator, on the same transport connection.

### 7.2.1 Content of ACCEPT SPDU

The ACCEPT SPDU contains

- a) a Connection Identifier parameter group, which is supplied by the called SS-user, to enable the SS-users to identify this specific session connection. This parameter group has no effect on the SPM. It contains
  - 1) a Called SS-user Reference parameter;
  - 2) a Common Reference parameter;
  - 3) an Additional Reference Information parameter.
- b) a Connect/Accept Item parameter group containing
  - 1) a Protocol Options parameter which allows the responder to indicate its ability to receive extended concatenated SPDUs;
  - 2) a TSDU Maximum Size parameter which, if present and not zero, indicates that segmenting of SSDUs has been proposed by the responder. The responder proposes alternative values for the maximum TSDU sizes for each direction of transfer (see 5.7.5). These values may be larger or smaller than the values supplied by the initiator in the CONNECT SPDU. The smaller value is used for the maximum TSDU size for each direction of transfer;
  - 3) a Version Number parameter to identify which versions of this protocol have been implemented. The highest version number indicated by both initiator and responder is used;
  - 4) an Initial Serial Number parameter which is present if the activity management functional unit is not selected and any of the minor synchronize, major synchronize or resynchronize functional units are selected regardless of whether or not the activity management functional unit is proposed. The called SS-user proposes the value, which is the value of the first serial number to be used;
  - 5) a Token Setting Item parameter supplied by the called SS-user, which indicates the initial token positions for each token available on this session connection, as derived from the selected functional units. A token is only available if any functional unit which requires that token has been selected for use on this session connection (see table 4), regardless of the settings of the token setting item parameter in the CONNECT SPDU (see 7.1.1b5)). If a token-controlled functional unit has been selected, then in the case where the calling SS-user has indicated that the initial assignment of the related token is the called SS-user's choice, this parameter contains a value chosen by the called SS-user. Otherwise, the values indicated by the calling SS-user in the CONNECT SPDU are selected and shall be returned.

c) a Token Item parameter which allows the called SS-user to request tokens which have been assigned to the calling SS-user in the CONNECT SPDU;

d) a Session User Requirements parameter which contains a list indicating the functional units proposed by the called SS-user and can be supported by the responder. The functional units selected for use on this session connection are the intersection of this set and the set proposed in the CONNECT SPDU (i.e. only those functional units indicated in both the CONNECT SPDU and the ACCEPT SPDU are selected). If both the half-duplex functional unit and the duplex functional unit were indicated in the CONNECT SPDU, then the ACCEPT SPDU shall propose which one is to be available. If only one of these functional units was indicated in the CONNECT SPDU, then the ACCEPT SPDU shall indicate that the same functional unit is to be used (or the connection attempt shall be rejected);

e) Calling Session Selector parameter corresponding to the calling SS-user may be present, in which case it will have the same value as in the CONNECT SPDU. Responding Session Selector parameter corresponding to the responding SS-user may be present and is derived from the Responding Session Address provided by the responding SS-user;

f) a User Data parameter which allows a limited amount of transparent user data to be passed from the called SS-user to the calling SS-user.

### 7.2.2 Sending the ACCEPT SPDU

An S-CONNECT (accept) response results in an ACCEPT SPDU. This SPDU is sent on the transport normal flow. After this successful connection, the SPM enters the data transfer phase and can receive any service request or SPDU that is allowed by the selected functional units and current token positions. If any of the minor synchronize, major synchronize or resynchronize functional units are selected but the activity management functional unit is not selected, the SPM sets V(A) and V(M) to the Initial Serial Number proposed by the called SS-user, which is the serial number to be used for the first synchronization point. V(R) is set to zero. Vsc is set false. If the activity management functional unit has been selected, Vact is set false.

### 7.2.3 Receiving the ACCEPT SPDU

A valid incoming ACCEPT SPDU results in an S-CONNECT (accept) confirm. After this successful connection, the SPM enters the data transfer phase and can receive any service request or SPDU that is allowed by the selected functional units and current token positions. If any of the minor synchronize, major synchronize or resynchronize functional units are selected but the activity management functional unit is not selected, the SPM sets V(A) and V(M) to the Initial Serial Number contained in the ACCEPT SPDU, which is the serial number to be used for the first synchronization point. V(R) is set to zero. Vsc is set false. If the activity management functional unit has been selected, Vact is set false.

If the called SS-user has requested any tokens in the Token Item parameter of the ACCEPT SPDU (see 7.2.1c)), an S-PLEASE-TOKEN indication is also generated.

## 7.3 REFUSE SPDU

A REFUSE SPDU is used by the responder to reject an attempt to establish a session connection.

### 7.3.1 Content of REFUSE SPDU

The REFUSE SPDU contains

a) a Connection Identifier parameter group, which is supplied by the called SS-user, to enable the SS-users to identify this specific session connection. This parameter group has no effect on the SPM. It contains

- 1) a Called SS-user Reference parameter;
- 2) a Common Reference parameter;
- 3) an Additional Reference Information parameter.

b) a Transport Disconnect Parameter which indicates whether or not the transport connection is to be kept;

c) a Session User Requirements parameter which contains a list of the functional units supported by the sending SPM, and required by the called SS-user;

d) a Version Number parameter to identify which versions of this protocol have been implemented by the sending SPM;

e) a Reason Code parameter giving the reason for refusal of the attempt to establish a session connection, together with a limited amount of transparent user data.

### 7.3.2 Sending the REFUSE SPDU

An S-CONNECT (reject) response results in a REFUSE SPDU. This SPDU is sent on the transport normal flow. No session connection is established. If the Transport Disconnect parameter indicates that the transport connection can be re-used, the SPM waits for a CONNECT SPDU. Otherwise, the SPM starts the timer, TIM, and waits for a T-DISCONNECT indication. If the timer expires before receipt of a T-DISCONNECT indication, the SPM requests transport disconnection with a T-DISCONNECT request. The timer is cancelled on receipt of a T-DISCONNECT indication.

NOTE — The value of TIM is a local implementation dependent matter, related to quality of service.

### 7.3.3 Receiving the REFUSE SPDU

A valid incoming REFUSE SPDU results in an S-CONNECT (reject) confirm. No session connection is established. If the Transport Disconnect parameter indicates that retention of the transport connection has been requested by the called SPM, and this is acceptable to the calling SPM, the SPM waits for an S-CONNECT request. Otherwise, the SPM releases the transport connection, by making a T-DISCONNECT request.

## 7.4 FINISH SPDU

Orderly release is initiated by transfer of a FINISH SPDU, which may be transferred during the data transfer phase. It requests as a response either:

- a) a DISCONNECT SPDU to complete the release of the session connection, or
- b) a NOT FINISHED SPDU to refuse the release of the session connection if the release token is available.

The FINISH SPDU is transferred in sequence with any normal data being transferred. The right to issue a FINISH SPDU is restricted to the owner of all available tokens.

### 7.4.1 Content of FINISH SPDU

The FINISH SPDU contains

- a) a Transport Disconnect parameter which indicates whether or not the transport connection is to be kept, subject to the restrictions specified in 6.2.4;
- b) a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.4.2 Sending the FINISH SPDU

An S-RELEASE request results in a FINISH SPDU. This SPDU is sent on the transport normal flow. After transferring a FINISH SPDU, the SPM may not send any further SPDUs (except ABORT SPDU or, in the case of collision of FINISH SPDUs, a DISCONNECT SPDU) unless a NOT FINISHED SPDU or a RESYNCHRONIZE SPDU is received, after which the data transfer phase may be resumed. Receipt of a DISCONNECT SPDU signals completion of orderly session release.

### 7.4.3 Receiving the FINISH SPDU

A valid incoming FINISH SPDU results in an S-RELEASE indication. The user data is passed to the SS-user. The SPM waits for an S-RELEASE response.

## 7.5 DISCONNECT SPDU

After receipt of a FINISH SPDU, a DISCONNECT SPDU may be transferred. Receipt of a DISCONNECT SPDU after transferring a FINISH SPDU signals the orderly release of the session connection. The DISCONNECT SPDU is transferred in sequence with any normal data being transferred.

### 7.5.1 Content of DISCONNECT SPDU

The DISCONNECT SPDU contains a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.5.2 Sending the DISCONNECT SPDU

An S-RELEASE (accept) response results in a DISCONNECT SPDU. This SPDU is sent on the transport normal flow. The session connection ceases to exist.

If the FINISH SPDU indicated that the transport connection is to be kept for reuse, and this is acceptable, the SPM waits for a CONNECT SPDU. Otherwise, the SPM starts the timer, TIM, and waits for a T-DISCONNECT indication. If the timer expires before receipt of a T-DISCONNECT indication, the SPM requests transport disconnection with a T-DISCONNECT request. The timer is cancelled on receipt of a T-DISCONNECT indication.

NOTE — The value of TIM is a local implementation dependent matter, related to quality of service.

### 7.5.3 Receiving the DISCONNECT SPDU

A valid incoming DISCONNECT SPDU results in an S-RELEASE (accept) confirm. The session connection ceases to exist.

If the transport connection is to be kept for reuse (see 6.2.4), the SPM waits for a suitable S-CONNECT request. Otherwise, a T-DISCONNECT request is issued.

#### NOTES

1 In the case of collision of a FINISH SPDU and an ABORT SPDU (see 7.7), the ABORT SPDU takes preference and thus the indication in the FINISH SPDU to keep or release the transport connection is ignored.

2 In the case of collision of FINISH SPDUs (data token and release token not available) the transport connection cannot be reused. The SPM receiving the DISCONNECT SPDU issues a T-DISCONNECT request.

## 7.6 NOT FINISHED SPDU

After receipt of a FINISH SPDU, a NOT FINISHED SPDU may be transferred subject to the token restrictions specified in table 5. No confirmation is sought.

### 7.6.1 Content of NOT FINISHED SPDU

The NOT FINISHED SPDU contains a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.6.2 Sending the NOT FINISHED SPDU

An S-RELEASE (reject) response results in a NOT FINISHED SPDU. This SPDU is sent on the transport normal flow. The SPM remains in the data transfer phase and can receive any service request or SPDU that is allowed by the available functional units and current token positions.

### 7.6.3 Receiving the NOT FINISHED SPDU

A valid incoming NOT FINISHED SPDU results in an S-RELEASE (reject) confirm. The SPM remains in the data transfer phase and can receive any service request or SPDU that is allowed by the available functional units and current token positions.

## 7.7 ABORT SPDU

The ABORT SPDU is used to reject a session connection establishment attempt, or to cause abnormal release of a

session connection at any time. This SPDU is also used by an SPM to release the session connection when a protocol error is detected. The ABORT SPDU may or may not request that the transport connection be released by the receiving SPM. Use of the ABORT SPDU may result in loss of data.

### 7.7.1 Content of ABORT SPDU

The ABORT SPDU contains

- a) a Transport Disconnect parameter which indicates whether or not the transport connection is to be kept;
- b) a Reflect Parameter Values parameter which, if present, allows implementation defined information to be transferred;
- c) a User Data parameter which, if present, allows a limited amount of transparent user data to be transferred.

### 7.7.2 Sending the ABORT SPDU

An S-U-ABORT request, or the detection of a protocol error in any state of the SPM, results in an ABORT SPDU. This SPDU is sent on the transport expedited flow, if it is available to this session connection. Otherwise this SPDU is sent on the transport normal flow. The SPM starts the timer, TIM, and waits for an ABORT ACCEPT SPDU or a T-DISCONNECT indication. Any other SPDUs are discarded. If the timer expires before receipt of an ABORT ACCEPT SPDU or a T-DISCONNECT indication, the SPM shall request transport disconnection with a T-DISCONNECT request. On receipt of a T-DISCONNECT indication, the timer is cancelled.

NOTE — The value of TIM is a local implementation dependent matter, related to quality of service.

### 7.7.3 Receiving the ABORT SPDU

A valid incoming ABORT SPDU results in an S-U-ABORT indication or an S-P-ABORT indication, depending on whether the abort is user generated or provider generated. The session connection ceases to exist. If the Transport Disconnect parameter in the received ABORT SPDU indicates that the transport connection is to be kept for reuse and this is acceptable to the receiving SPM, an ABORT ACCEPT SPDU is sent. If the Transport Disconnect parameter in the received ABORT SPDU indicates that the transport connection is not to be kept for reuse or reuse of the transport connection is not acceptable to the receiving SPM, the receiving SPM either

- a) releases the transport connection, or
- b) sends an ABORT ACCEPT SPDU (see 7.8).

Receiving an ABORT SPDU sent in response to a CONNECT SPDU results in

- a) a T-DISCONNECT request, unless retention of the transport connection has been requested in the ABORT SPDU, in which case the ABORT SPDU is acknowledged with an ABORT ACCEPT SPDU (see 7.8), and
- b) an S-P-ABORT indication or an S-U-ABORT indication to the SS-user.

## 7.8 ABORT ACCEPT SPDU

The ABORT ACCEPT SPDU is used to return a confirmation to the ABORT SPDU.

### 7.8.1 Content of ABORT ACCEPT SPDU

The ABORT ACCEPT SPDU contains no parameters.

### 7.8.2 Sending the ABORT ACCEPT SPDU

A valid incoming ABORT SPDU results in sending an ABORT ACCEPT SPDU, when the transport connection can be reused, i.e. when

- a) the transport expedited service is not available to this session connection, and
- b) retention of the transport connection has been requested in the ABORT SPDU and it is acceptable to reuse the transport connection.

The SPM, as a local implementation decision, may send an ABORT ACCEPT SPDU in response to an ABORT SPDU, even if the transport connection is not to be kept.

This SPDU is sent on the transport expedited flow, if it is available to this session connection. Otherwise, this SPDU is sent on the transport normal flow. The session connection ceases to exist.

### 7.8.3 Receiving the ABORT ACCEPT SPDU

A valid incoming ABORT ACCEPT SPDU results in resetting the timer, TIM, and

- a) releasing the transport connection, if release of the transport connection was requested in the previously sent ABORT SPDU;
- b) if retention of the transport connection was requested, the transport connection is now available for reuse by a new session connection, if this SPM was the initiator of the transport connection (see 6.1).

The session connection ceases to exist.

## 7.9 DATA TRANSFER SPDU

Normal data is transferred by use of the DATA TRANSFER SPDU. If the extended concatenation option was selected during connection establishment, certain concatenations of the DATA TRANSFER SPDU with other SPDUs are allowed (see 6.3.7).

The right to issue a DATA TRANSFER SPDU is subject to the token restrictions specified in table 5.

### 7.9.1 Content of DATA TRANSFER SPDU

The DATA TRANSFER SPDU contains

a) an Enclosure Item parameter to indicate the beginning and end of the SSDU when segmenting has been selected. When segmenting has been selected, the Enclosure Item parameter is always present and indicates whether the SPDU is the beginning, middle or end of the SSDU. When segmenting has not been selected, the Enclosure Item parameter is not present;

b) a User Information Field to transfer transparent user data whose maximum size is unlimited when segmenting has not been selected and whose maximum size is limited by the maximum TSDU size when segmenting has been selected.

### 7.9.2 Sending the DATA TRANSFER SPDU

An S-DATA request results in a DATA TRANSFER SPDU unless segmenting has been selected, in which case an ordered sequence of DATA TRANSFER SPDUs will be sent with the appropriate value for the Enclosure Item parameter until the complete SSDU has been transferred.

The concatenation of any segment of an SSDU with any other SPDU will not result in a TSDU larger than the selected maximum TSDU size for that direction of transfer. However, there is no requirement that the resulting TSDU should be of the maximum size for that direction of transfer. All DATA TRANSFER SPDUs, except the last DATA TRANSFER SPDU in a sequence greater than one, must have user information. DATA TRANSFER SPDUs are sent on the transport normal flow.

### 7.9.3 Receiving the DATA TRANSFER SPDU

A valid incoming DATA TRANSFER SPDU results in an S-DATA indication unless segmenting has been selected. In this case, a valid incoming DATA TRANSFER SPDU, which indicates end of SSDU, results in an S-DATA indication to pass the entire SSDU to the SS-user.

Where segmenting has been selected and an incomplete segmented SSDU is outstanding, the receipt of

RESYNCHRONIZE SPDU  
EXCEPTION REPORT SPDU  
EXCEPTION DATA SPDU  
ACTIVITY INTERRUPT SPDU  
ACTIVITY DISCARD SPDU  
ABORT SPDU  
PREPARE (RESYNCHRONIZE) SPDU

has a destructive effect on the entire SSDU (i.e. the SPDUs which have already been received are discarded, the remaining SPDUs will not be received).

It is also valid to receive EXPEDITED SPDUs and PREPARE (MAJOR SYNC ACK) SPDUs and this has no effect on the segmented SSDU being received.

The receipt of any other SPDUs is a protocol error.

## 7.10 EXPEDITED SPDU

The EXPEDITED SPDU is used to transfer expedited SSDUs.

The right to send expedited data is not associated with any tokens. When this functional unit is selected, both SS-users may send expedited data. An expedited SSDU may be delivered to the receiving SS-user prior to other SSDUs previously transferred on the transport normal flow; it may not be delivered to the receiving SS-user later than any SSDUs transferred after it.

Expedited SSDUs are delivered to the receiving SS-user in the same sequence in which they were issued by the sending SS-user.

### 7.10.1 Content of EXPEDITED SPDU

The EXPEDITED SPDU contains a User Information field which allows a limited amount of transparent user data to be transferred.

### 7.10.2 Sending the EXPEDITED SPDU

An S-EXPEDITED-DATA request results in an EXPEDITED SPDU being sent. This SPDU is sent on the transport expedited flow.

### 7.10.3 Receiving the EXPEDITED SPDU

A valid incoming EXPEDITED SPDU results in an S-EXPEDITED-DATA indication.

## 7.11 TYPED DATA SPDU

The TYPED DATA SPDU enables the SS-users to transmit transparent user data, irrespective of the availability or assignment of the data token. In all other respects, the same constraints apply as for normal data (see 7.9). The same rules for segmenting also apply.

### 7.11.1 Content of TYPED DATA SPDU

The TYPED DATA SPDU contains

a) an Enclosure Item parameter to indicate the beginning and end of SSDU when segmenting has been selected. When segmenting has been selected, the Enclosure Item parameter is always present and indicates whether the SPDU is the beginning, middle or end of the SSDU. When segmenting has not been selected, the Enclosure Item parameter is not present;

b) a User Information Field to transfer transparent user data whose maximum size is unlimited when segmenting has not been selected and whose maximum size is limited by the maximum TSDU size when segmenting has been selected.

### 7.11.2 Sending the TYPED DATA SPDU

An S-TYPED-DATA request results in the transfer of a TYPED DATA SPDU unless segmenting has been selected, in which case an ordered sequence of TYPED DATA SPDUs will be sent with the appropriate value for the Enclosure Item parameter until the complete SSDU has been transferred. Each SPDU is mapped onto one TSDU and will not be larger than the selected

maximum TSDU size for that direction of transfer. However, there is no requirement that the resulting TSDU should be of the maximum size for that direction of transfer. All TYPED DATA SPDUs, except the last TYPED DATA SPDU in a sequence greater than one, must have user information. TYPED DATA SPDUs are sent on the transport normal flow. The current state of the SPM is not changed.

### 7.11.3 Receiving the TYPED DATA SPDU

A valid incoming TYPED DATA SPDU results in an S-TYPED-DATA indication, unless segmenting has been selected. In this case, a valid incoming TYPED DATA SPDU which indicates the end of SSDU results in an S-TYPED-DATA indication to pass the entire SSDU to the SS-user. The current state of the SPM is not changed.

When segmenting has been selected, the rules governing the receipt of SPDUs other than TYPED DATA SPDUs during the transfer of a segmented SSDU are the same as for the DATA TRANSFER SPDU (see 7.9.3).

## 7.12 CAPABILITY DATA SPDU

The CAPABILITY DATA SPDU is used to transfer a limited amount of transparent user data outside activities (i.e. when the activity management functional unit has been selected and Vact is false). The right to send this SPDU is restricted to the side having the right to start the next activity (i.e. the activity management functional unit has been selected and Vact is false and subject to the token restrictions specified in table 5).

### 7.12.1 Content of CAPABILITY DATA SPDU

The CAPABILITY DATA SPDU contains a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.12.2 Sending the CAPABILITY DATA SPDU

An S-CAPABILITY-DATA request results in a CAPABILITY DATA SPDU being sent. This SPDU is sent on the transport normal flow. The SS-user is not permitted to issue a further S-CAPABILITY-DATA request until this CAPABILITY DATA SPDU is acknowledged.

### 7.12.3 Receiving the CAPABILITY DATA SPDU

A valid incoming CAPABILITY DATA SPDU results in an S-CAPABILITY-DATA indication to the SS-user.

## 7.13 CAPABILITY DATA ACK SPDU

The CAPABILITY DATA ACK SPDU is used to complete the capability data exchange.

### 7.13.1 Content of CAPABILITY DATA ACK SPDU

The CAPABILITY DATA ACK SPDU contains a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.13.2 Sending the CAPABILITY DATA ACK SPDU

The SS-user generates an S-CAPABILITY-DATA response which results in a CAPABILITY DATA ACK SPDU. This SPDU is sent on the transport normal flow.

### 7.13.3 Receiving the CAPABILITY DATA ACK SPDU

A valid incoming CAPABILITY DATA ACK SPDU results in an S-CAPABILITY-DATA confirm. This allows the SS-user to issue a further S-CAPABILITY-DATA request.

## 7.14 GIVE TOKENS SPDU

The GIVE TOKENS SPDU is used

- a) to introduce a concatenated sequence of SPDUs; and/or
- b) to cause assignment of currently owned tokens to be changed.

If the GIVE TOKENS SPDU does not contain a parameter field, it is used to indicate concatenation without assignment of tokens and, in this case, the sending and receiving procedures do not apply.

### 7.14.1 Content of GIVE TOKENS SPDU

The GIVE TOKENS SPDU contains a Token Item parameter which indicates which tokens are being transferred from the sending SS-user to the receiving SS-user.

### 7.14.2 Sending the GIVE TOKENS SPDU

An S-TOKEN-GIVE request results in a GIVE TOKENS SPDU. This SPDU is sent on the transport normal flow.

### 7.14.3 Receiving the GIVE TOKENS SPDU

A valid incoming GIVE TOKENS SPDU results in an S-TOKEN-GIVE indication.

## 7.15 PLEASE TOKENS SPDU

The PLEASE TOKENS SPDU is used

- a) to introduce a concatenated sequence of SPDUs; and/or
- b) to request that the token assignments be changed to permit the requestor to be authorized to perform a function associated with the requested tokens.

If the PLEASE TOKENS SPDU does not contain any parameter fields, it is used to indicate concatenation without requesting tokens and, in this case, the sending and receiving procedures do not apply.

### 7.15.1 Content of PLEASE TOKENS SPDU

The PLEASE TOKENS SPDU contains

- a) a Token Item parameter which indicates which tokens are being requested by the sending SS-user;
- b) a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.15.2 Sending the PLEASE TOKENS SPDU

An S-TOKEN-PLEASE request results in a PLEASE TOKENS SPDU. This SPDU is sent on the transport normal flow.

### 7.15.3 Receiving the PLEASE TOKENS SPDU

A valid incoming PLEASE TOKENS SPDU results in an S-TOKEN-PLEASE indication. Receiving a PLEASE TOKENS SPDU for tokens which are not currently assigned to the accepting SS-user is not a protocol error.

## 7.16 GIVE TOKENS CONFIRM SPDU

The GIVE TOKENS CONFIRM SPDU is used as a result of an S-CONTROL-GIVE request to cause assignment of all of the currently assigned tokens to be changed when Vact is false. Receipt of the GIVE TOKENS CONFIRM SPDU by the receiving SPM is acknowledged by the GIVE TOKENS ACK SPDU.

### 7.16.1 Content of GIVE TOKENS CONFIRM SPDU

The GIVE TOKENS CONFIRM SPDU contains no parameters.

### 7.16.2 Sending the GIVE TOKENS CONFIRM SPDU

An S-CONTROL-GIVE request when Vact is false results in a GIVE TOKENS CONFIRM SPDU. The SPM then waits for a GIVE TOKENS ACK SPDU before permitting further SPDUs, associated with the available tokens, to be sent or received. SPDUs not associated with tokens (for example TYPED DATA SPDU) may be sent or received as normal. This SPDU is sent on the transport normal flow.

### 7.16.3 Receiving the GIVE TOKENS CONFIRM SPDU

A valid incoming GIVE TOKENS CONFIRM SPDU results in an S-CONTROL-GIVE indication, followed by a GIVE TOKENS ACK SPDU.

## 7.17 GIVE TOKENS ACK SPDU

The GIVE TOKENS ACK SPDU is used to acknowledge receipt of a GIVE TOKENS CONFIRM SPDU. The GIVE TOKENS ACK SPDU can only be sent when Vact is false.

### 7.17.1 Content of GIVE TOKENS ACK SPDU

The GIVE TOKENS ACK SPDU contains no parameters.

### 7.17.2 Sending the GIVE TOKENS ACK SPDU

A valid incoming GIVE TOKENS CONFIRM SPDU results in a GIVE TOKENS ACK SPDU (see also 7.16.3). The SPM may now transmit SPDUs associated with the token controlled functional units. This SPDU is sent on the transport normal flow.

### 7.17.3 Receiving the GIVE TOKENS ACK SPDU

After receiving a valid incoming GIVE TOKENS ACK SPDU, the SPM is now prepared to receive any SPDUs associated with the token controlled functional units.

## 7.18 MINOR SYNC POINT SPDU

The MINOR SYNC POINT SPDU is used to define a minor synchronization point. A confirmation may be returned by the receiver but is not required by the SPM (see 7.19). All acknowledgement rules are defined by the SS-users. In particular, whether confirmation is requested or not is transparent to the SPM. The right to issue a MINOR SYNC POINT SPDU is subject to the token restrictions specified in table 5.

### 7.18.1 Content of MINOR SYNC POINT SPDU

The MINOR SYNC POINT SPDU contains

- a) a Sync Type Item parameter which is used to indicate if an explicit confirmation is required (see 7.19);
- b) a Serial Number parameter which indicates the serial number of this minor synchronization point, and is set by the SPM to the current value of V(M);
- c) a User Data parameter which allows a limited amount of transparent user data to be transferred.

### 7.18.2 Sending the MINOR SYNC POINT SPDU

An S-SYNC-MINOR request results in a MINOR SYNC POINT SPDU. This SPDU is sent on the transport normal flow. If Vsc is true, V(A) is set equal to V(M) and Vsc is set false. V(M) is incremented by one.

### 7.18.3 Receiving the MINOR SYNC POINT SPDU

A valid incoming MINOR SYNC POINT SPDU results in an S-SYNC-MINOR indication. If Vsc is false, V(A) is set equal to V(M) and Vsc is set true. V(M) is incremented by one.

## 7.19 MINOR SYNC ACK SPDU

The MINOR SYNC ACK SPDU is used to return a confirmation to minor synchronization points. The SPM sends a MINOR SYNC ACK SPDU for each S-SYNC-MINOR response.