Draft EN 301 069-1 V1.1.1 (1997-08)

European Standard (Telecommunications series)

Integrated Services Digital Network (ISDN); Signalling System No.7; ISDN User Part (ISUP); Application Transport Mechanism (APM) Part 1: Protocol specification

European Telecommunications Standards Institute

Reference
DEN/SPS-01042-1 (9wc90ico.PDF)

Keywords
APM, ISDN, ISUP, NNI, SS7

ETSI Secretariat

Postal address
F-06921 Sophia Antipolis Cedex - FRANCE

Office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

X.400
c= fr; a=atlas; p=etsi; s=secretariat

Internet
secretariat@etsi.fr
http://www.etsi.fr

Contents

Intellectual Property Rights

ETSI has not been informed of the existence of any Intellectual Property Right (IPR) which could be, or could become essential to the present document. However, pursuant to the ETSI Interim IPR Policy, no investigation, including IPR searches, has been carried out. No guarantee can be given as to the existence of any IPRs which are, or may be, or may become, essential to the present document.

Foreword

This European Standard (Telecommunication series) has been produced by ETSI Technical Committee Signalling Protocols and Switching (SPS), and is now submitted for the Public Enquiry phase of the ETSI standards Two-step Approval Procedure (TAP).

The present document is part 1 of a multi-part EN covering Integrated Services Digital Network (ISDN); Signalling System No.7; ISDN User Part (ISUP); Application Transport Mechanism (APM) as identified below:

Part 1: "Protocol specification";

Part 2: "Protocol Implementation Conformance Statement (PICS) proforma";

Part 3: "Test Suite Structure and Test Purposes (TSS&TP) specification";

Part 4: "Abstract Test Suite (ATS) specification".

Proposed national transposition dates	
Date of latest announcement of this EN (doa):	3 months after ETSI publication
Date of latest publication of new National Standard or endorsement of this EN (dop/e):	6 months after doa
Date of withdrawal of any conflicting National Standard (dow):	6 months after doa

1 Scope

The present document describes the additions to the ISDN User Part (ISUP) for the introduction of a transport mechanism for use by applications requiring a bearer in conjunction with the support of the application's signalling information flow. This transport mechanism provides the same capabilities as the Transaction Capabilities Application Part (TCAP) provides to its users. The definition of the applications that use this mechanism are documented in their own standards and are therefore beyond the scope of the present document.

The Application Transport Mechanism (APM) is capable of creating signalling associations between APM-user application logic located at a public initiating node (PIN) and its peer APM-user application logic located at the public addressed node (PAN) (the PIN/PAN concept is described in clause 5). The ability to create different PIN/PAN relationships is only limited by the addressing mechanism used. At call setup, the ISUP basic call mechanism is used to provide the addressing for the APM; consequently it will route a call through the network towards the PAN in the same manner as it normally establishes a call.

At any other time of the call, it is possible to create the following PIN/PAN relationships:

a) PIN and PAN at end exchanges;

b) PIN at transit exchange and PAN at end exchange;

c) PIN at end exchange and PAN at transit exchange;

d) PIN and PAN at transit exchanges.

This means that a PIN located anywhere in the call path has the capability to create a PIN/PAN relationship with the next exchange in the call path having the APM-user capability. The intermediate exchanges having the APM capability but not the APM-user capability will transit the APM information.

It is, of course, possible to communicate in both directions once a PIN/PAN relationship has been created. It should be noted that only one PIN/PAN relationship may exist over the same segment of the call path for a given application (Context identity value). For example, if a PIN/PAN relationship has been established between the originating local exchange (OLE) and an intermediate exchange in the call path (PAN), it is not possible to later establish a PIN/PAN relationship from the destination local exchange to the originating local exchange for the same application since its corresponding PAN would be located at the OLE and would therefore result in two PIN/PAN relationships overlapping at the call leg between the OLE and the intermediate exchange.

2 Normative references

References may be made to:

a) specific versions of publications (identified by date of publication, edition number, version number, etc.), in which case, subsequent revisions to the referenced document do not apply; or

b) all versions up to and including the identified version (identified by "up to and including" before the version identity); or

c) all versions subsequent to and including the identified version (identified by "onwards" following the version identity); or

d) publications without mention of a specific version, in which case the latest version applies.

A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

[1] EN 300 356-1 (not yet available): "Integrated Services Digital Network (ISDN); Signalling System No.7; ISDN User Part (ISUP) version 2 for the international interface; Part 1: Basic services [ITU-T Recommendations Q.761 to Q.764 (1993), modified]".

[2] ITU-T Recommendation Q.1400 (1993): "Architecture framework for the development of signalling and OA&M protocols using OSI concepts".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following definition applies:

context: A set of rules by which to interpret the contents of the Encapsulated Application Information field as identified by a context identifier which uniquely represents an application with which data is associated.

In addition, the definitions of EN 300 356-1 [1] apply.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AE	Application Entity
AEI	Application Entity Instance
ALS	Application Layer Structuring
AP	Application Process
APM	Application Transport Mechanism
APP	Application Transport Parameter
ASE	Application Service Element
ATII	Application Transport Instruction Indicators
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
M/O	Mandatory/Optional
MACF	Multiple Association Control Function
MTP-3	Message Transfer Part
NI	Network Interface
NNI	Network Node Interface
OLE	Originating Local Exchange
OSI	Open Systems Interconnection
PAN	Public Addressed Node
PIN	Public Initiating Node
PRI	Pre-Release Information message
SACF	Single Association Control Function
SAO	Single Association Object
SDL	Specification and Description Language
SID	Signalling Identifier
SIO	Service Indicator Octet
SLS	Signalling Link Set
TCAP	Transaction Capabilities Application Part
UCEH	Unidentified Context and Error Handling

4 Document structure

The description of the ISDN User Part procedures in the present document are structured according to the model described in subclause 5.2. The description is thus divided into two main parts:

- Protocol functions.

- Non-protocol functions, i.e. exchange nodal functions; this is referred to as the "Application Process".

NOTE: The present document describes only the part of the total Application Process and Protocol functions in the exchange that relates to Network Node Interface (NNI) enhancements for the support of the APM.

The protocol functions relate only to signalling associations with a bearer (ISUP) and describe the additional information and procedures to that defined for ISUP Basic Call (EN 300 356-1 [1]) and its associated formats and codes required to support the APM.

The signalling association with a bearer is subdivided into three parts: APM-user Protocol Control (APM-user ASE), Application Transport Mechanism protocol control (APM ASE), and the ISUP Basic Call (ISUP ASE). These are co-ordinated by the Single Association Co-ordination Function (SACF).

The Application Entity (AE) can contain more than one APM-user application. It also contains Unidentified Context and Error Handling (UCEH) Application Service Element (ASE) for taking the appropriate actions in the case when the addressed node (PAN) does not support the APM-user application or for the handling of APM error cases.

The Application Process (AP) contains all Call Control functions, however, the present document will only describe the enhancements required to support the Application Transport Mechanism (APM). The AP relevant to the public ISUP basic call can be found within EN 300 356-1 [1], whereas the APM-user functionality can be found in the suitable standards/recommendations that define the given application.

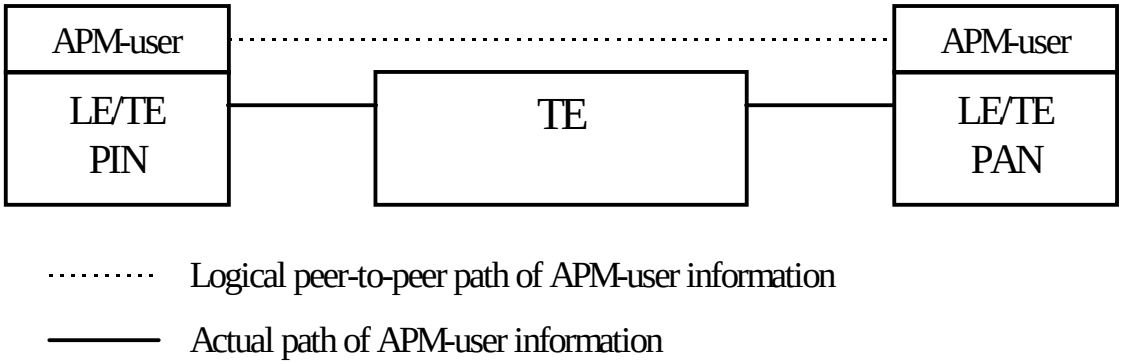
The service primitive technique, used to define the ISUP ASEs and the SACF specific to the Application Transport Mechanism's signalling needs, is a way of describing how the services offered by an ASE, or SACF, - the provider of a (set of) service(s) - can be accessed by the user of the service(s) - the SACF or the AP, respectively.

The service primitive/ interface is a conceptual interface and is not a testable or accessible interface. It is a descriptive tool. The use of service primitives at an interface does not imply any particular implementation of that interface, nor does it imply that an implementation needs to conform to that particular service primitive interface to provide the stated service. All conformance to the ISUP specification is based on the external behaviour of a node, i.e. on the generation of the correct message structure and in the proper sequence (as specified in EN 300 356-1 [1] and the present document).

5 Modelling

The models described in this clause introduce concepts and terminology used in the present document of the APM.

5.1 Network model



APM-user	User Application of the Application Transport Mechanism
LE	Local public Exchange
TE	Transit public Exchange
PIN	Public Initiating Node - Initiating APM functionality
PAN	Public Addressed Node - Addressed APM functionality

Figure 1: Network model

This subclause illustrates the relationship between the APM-user Application and the APM implemented across a public network. Figure 1 provides an example of a call from an initiating application to an addressed application via a public transit network. The public network (APM) provides the APM-user application with the service of providing both a signalling association as well as a bearer between remote applications.

The Public Initiating Node (PIN) and Public Addressed Node (PAN) concept is introduced here to assist in the description of the APM. The PIN represents the point in the network that wishes to initiate communications towards a peer APM-user application located at an addressed location (PAN). An APM-user application may result in the establishment of a signalling and bearer association in which case it will use the services of the public basic call.

A PIN/PAN signalling relationship may be established at any time in association with an already established public basic call, or an application may initiate the establishment of a new call. In the latter case, the initiating APM-user's Application Process supplies a normal public E.164 number which is used to route through the public network, thus establishing an association between the public initiating node (PIN) and the public addressed node (PAN). The PAN identifies the particular APM-user application by the Context Identifier value carried within the Application Transport Parameter (APP).

The public basic call mechanism is employed to provide an association between the PIN and the PAN. In routeing through the public network, the call may pass through intermediate public nodes with or without the ability to support the particular APM-user functionality; however as the application is not addressed to that node, it will behave as a normal intermediate public node.

As a number of APM-user applications may be independently active on the same basic call, it is possible to have a number of PIN/PAN relationships established in either the forward or backward direction and at different phases with respect to the ISUP basic call.

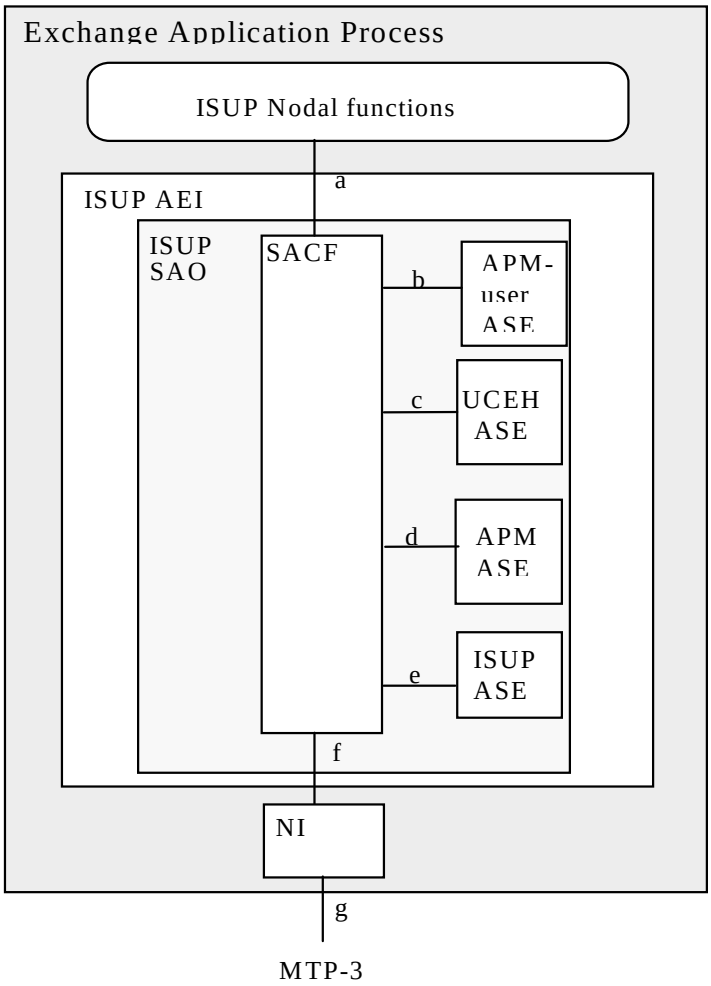
5.2 Specification model

5.2.1 Introduction

The model used to structure the description of ISUP procedures herein are based on the Open Systems Interconnection (OSI) Application Layer Structure (ALS) model, see ITU-T Recommendation Q.1400 [2]. This subclause presents the model and gives a general description of its operation. Subclause 5.2 shows the generalized model for the "Exchange Application Process" for the support of applications making the use of the Application Transport Mechanism over the Network Node Interface (NNI).

5.2.2 General Model

The generalized model for the bearer related (ISUP) APM AP is shown in figure 2. This figure does not represent the situation at any specific point during ISUP procedures, but instead it shows the full picture of the architecture. The specific application of this model is discussed in subclause 5.1/Q.apm. Figure 2 shows the primitive interfaces between the functional blocks, as used in the body of the present document for calls with bearer (ISUP).



Abbreviations:

AEI	Application Entity Invocation
ASE	Application Service Element
APM	Application Transport Mechanism
APM-user	Application Transport Mechanism user application
ISUP	ISDN User Part
MTP	Message Transfer Part
NI	Network Interface
SACF	Single Association Control Function
SAO	Single Association Object
UCEH	Unidentified Context and Error Handling

Figure 2: ISUP specification model

The definition of the interfaces a to g are:

a) interface between the Application Process nodal functions (AP) and the SACF for the support of the APM over the NNI: see subclause 6.2.2;

b) interface between SACF and APM-user ASE representing the application being supported by the APM. (interface is beyond the scope of the present document, *except for the primitive interface part used to access the services of the APM ASE*): see subclause 11.1;

c) interface between SACF and UCEH ASE representing the handling of unidentified context identifier values and error cases: see subclause 10.1;

d) interface between SACF and APM ASE representing enhancements of the public functionality (ISUP) for providing a transportation mechanism for the support of various applications (APM-user) over the NNI: see subclause 9.1;

e) interface to public ISUP basic call signalling ASE (interface is beyond the scope of the present document, *except for the primitive interface part used by the APM ASE to access the services of the ISUP ASE*): see clause 8;

f) interface between SACF and NI function: see subclause 12.2.2;

g) interface to MTP-3.

With respect to figure 2, all functions also have an interface to a "Management application"; this is not defined as a formal primitive interface.

The term "Exchange Application Process" is used to describe all the Application functionality in an exchange. ISUP is a part of the Exchange Application Process. Thus the ISUP nodal functions shown on the model are referred to as the ISUP Application Process functions in the body of the present document.

The ISUP Application Entity Instance (AEI) provides all the bearer related communication capabilities required by the ISUP nodal functions. For simplicity an ISUP AEI is defined as containing just one Single Association Object (SAO); this avoids the need to specify a Multiple Association Control Function (MACF). Thus all co-ordination of ISUP signalling associations are performed via the ISUP Nodal functions.

The SACF has the responsibility of co-ordinating the flow of primitives between its interfaces in the appropriate manner.

The ISUP ASE is defined in by EN 300 356-1 [1]. Its main responsibilities are basic call procedures and the handling of protocol errors and unrecognized information handling (EN 300 356-1 [1]). The monolithic nature of these recommendations means that both Public Call Control and Protocol Control functionality are defined together. It is not the intention of the present document to re-define EN 300 356-1 [1] in ALS format, therefore it is referenced en-bloc within the present document as the ISUP ASE. Conceptually, this should be considered to represent a logical division between the protocol control functionality within the ISUP ASE and its associated call control functionality within the application process. The modelling and interfaces with respect to this are beyond the scope of the present document.

The APM ASE provides the means for the transfer of information between nodes for signalling requiring a bearer, and for providing generic services to applications, while being independent of any of these. It is responsible for the enhancements to the NNI (ISUP) for the support of a mechanism which allows various applications to transport their information flows via the NNI. Its main responsibility is to provide message segmentation in order to provide the APM-user the ability to transport up to 2 048 octets of application information. The APM ASE is able to support multiple APM-users where each is treated independently and provided with the same level of service.

The UCEH ASE handles the Application Transport Instruction Indicators (ATII), thereby providing a compatibility mechanism for the case where various levels of application (context) support exists within network nodes. It also handles APM reassembly error cases according to the ATII. For the compatibility mechanism it consists of two distinct sets of functions; one set used as the PAN, and one set used as the PIN (supporting the signalling association to a subsequent exchange). The PIN/PAN concept is explained in subclause 5.1.

The APM-user ASE is responsible for preparing the application specific signalling in a form that can be transported by the public APM.

To handle any particular ISUP function the Exchange Application Process creates an instance of the required ISUP nodal functions. ISUP will create instances, as required, of the ISUP AEI. The Network Interface (NI) function exists to distribute messages received from the Message Transfer Part (MTP-3) to the appropriate instance of the ISUP AEI. There is only one instance of the NI in an exchange. The NI also fulfils the presentation layer function according to the formats and codes of EN 300 356-1 [1].

The MTP-3 interface is defined in EN 300 356-1 [1].

The SAO contained in the ISUP AE is one of the following types:

a) Public Initiating node (PIN)

   This contains:

   - ISUP, Initiating APM, Initiating UCEH ASE, Initiating APM-user and ISUP SACF.

b) Public addressed node (PAN)

This contains:

- ISUP, Addressed APM, Addressed UCEH ASE, Addressed APM-user and ISUP SACF.

The ISUP may be incoming or outgoing depending on the direction of the PIN/PAN relationship with respect to the ISUP call.

5.2.3 Dynamic primitive flows

Figures 3 and 4 illustrate the dynamic primitive flows for a APM-user call with a bearer being supported over the NNI (ISUP) for the case that a call control message is coincident with the application information flow. Figure 3 shows the case when a message is being sent, figure 4 shows the case when a message is being received.

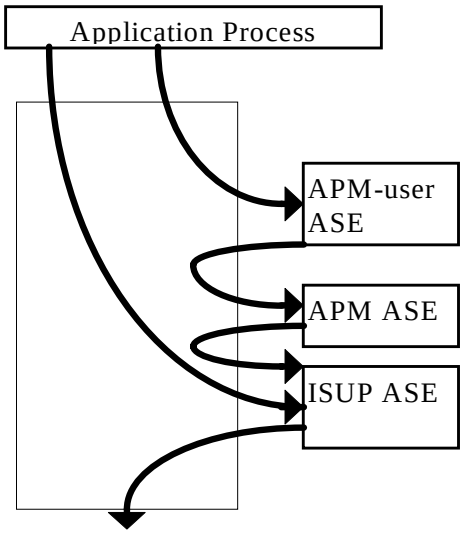


Figure 3

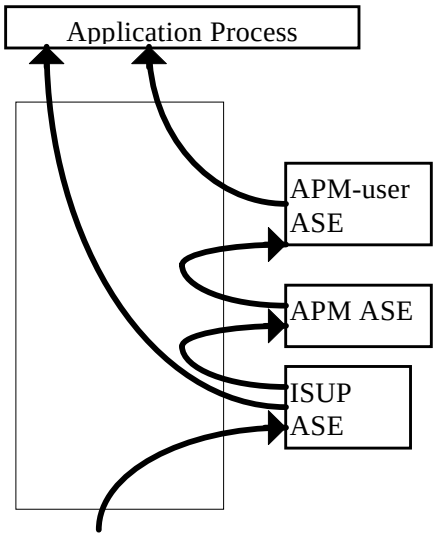


Figure 4

Figures 5 and 6 illustrate the dynamic primitive flows for the NNI support of the APM-user information flow via a call with a bearer where no call control messages are sent coincidently. That is, the APM ASE initiates a primitive towards the ISUP ASE which in turn sends an APM message which will provide a mechanism for supporting the information flow.

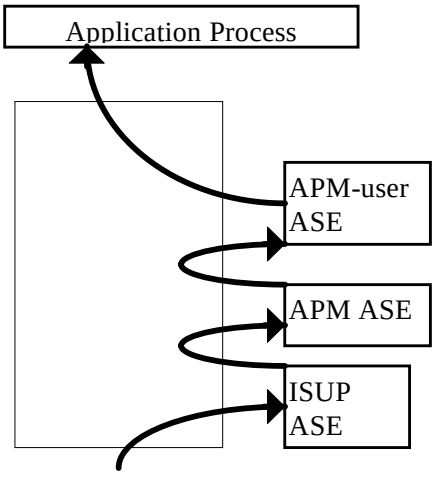


Figure 5

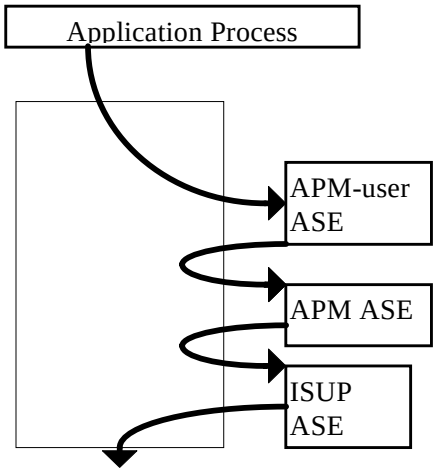


Figure 6

When the APM ASE and associated AP logic needs to communicate, primitives are passed between them. Figure 7 illustrates the case when the AP requests a service directly from the APM ASE. Figure 8 illustrates an indication being sent directly from the APM ASE to the AP.

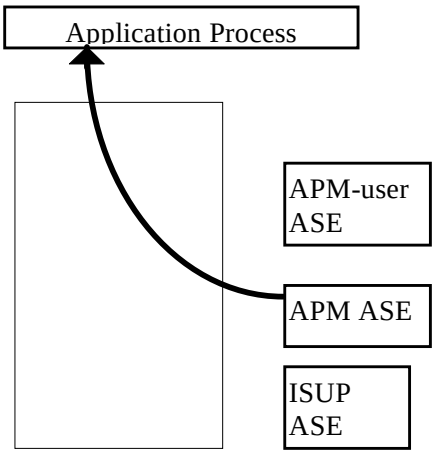


Figure 7

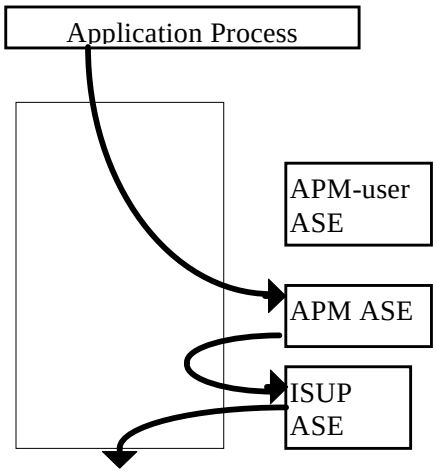


Figure 8

6 Application Process functions

6.1 General

The modelling of the Application Process (AP) is beyond the scope of the present document, however, in order to appreciate the role of the AP for the purposes of the present document, it can be considered to consist of three different functional groups that are relevant to the support of an application over the public NNI. These are the APM-user application processes (beyond the scope of the present document), the public network ISUP basic call functionality, where the application process and protocol control functions are defined together in the monolithic recommendations of the ISUP (EN 300 356-1 [1]), and the APM specific application process logic which is defined in the present document.

It is the responsibility of the APM-user aspect of the application process to co-ordinate between the APM-user application process and the public basic call application process functionality. It has to ensure that the basic call states are aligned between the public basic call and the APM-user application processes.

When the APM-user application requires the establishment of a PIN/PAN signalling association and the related ISUP call already exists, the application makes use of the APM services in conjunction with the existing call. When no call already exists, it passes the routeing information to the public application process such that it can route the call from the PIN to the appropriate node in the public network, PAN, which contains the adjacent APM-user functionality. The PIN/PAN concept is described in subclause 5.1. Details of the Public basic call routeing information requirements can be found in EN 300 356-1 [1].

The definition of the primitive interface (interface a in figure 2) between the AP and the SACF for both the specific public basic call primitives and those of the APM-user application are beyond the scope of the present document.

6.2 APM Application Process (AP) functions

6.2.1 Introduction

The following functions are fulfilled by the APM application process.

Transit node operation

The public network may contain a number of transit exchanges between the PIN and the PAN. Some or all of these exchanges may support the APM, however they are not the addressed node (PAN) for the application using the APM. In this case these nodes needs to pass the APM information transparently with the behaviour of a transit exchange.

Support of APM signalling association acknowledgement mechanism (sent from the PAN in conjunction with the APM segmentation/reassembly procedures)

When sending information in subsequent messages after an IAM but before the reception of the first backwards message, there is no guarantee that the information will successfully arrive at its destination. Since APM-user applications may wish to send a greater amount of information during the establishment phase than can be carried in an ISUP IAM message, it is necessary to send the information in subsequent messages. To ensure the reliable delivery of this information, the application process located at the PAN shall initiate the sending of the appropriate first backwards message (such as an ACM with "no indication" or APM) towards the PIN on reception of an IAM indicating segmentation of APM information has occurred. This provides an acknowledgement to the IAM, thereby confirming the establishment of a signalling association.

The co-ordination of segmented APM-user information with public call control messages

When application data is segmented by the APM, the result is that the information is sent over the ISUP via a number of messages. It is therefore the responsibility of the APM to ensure that the information is reassembled correctly and that the application data is associated with the correct call control message, where applicable. This shall be achieved by the APM ASE informing the AP that the complete information has not been made available to the AP (i.e. reassembly procedure underway). Similarly the AP shall be informed once the complete information has been passed to it, thus allowing it to complete its processing. It is the responsibility of the AP to co-ordinate all the application processes related to the call.

The handling of instructions received from the SACF resulting from the handling of Application Transport Instruction Indicators

The ATII mechanism allows an APM-user application to instruct a PAN how to respond to error cases; for example it may wish to release the call in the case where the application is not supported at the PAN. This instruction is fulfilled when the AP receives the UCEH_Release indication primitive.

Use of Pre-release message

The ISUP RELEASE message does not support the transport of the APP through exchanges using a previous version of ISUP. Any APM-user application requiring the transport of application data in the release phase of a call has to make use of the pre-release mechanism (PRI message) described in EN 300 356-1 [1].

6.2.2 Primitive interface (AP - SACF)

The primitives in table 1 have been introduced by the functionality specified in the present document. All other primitives at this interface (public basic call or APM-user application) are beyond the scope of the present document.

Table 1: Primitives between AP and SACF

Primitive name	Types
APM_Transit	Indication/Request
APM_Acknowledgement	Request
More_APP_Info	Indication
End_APP_Info	Indication
UCEH_Release	Indication

6.2.3 Procedures

This subclause describes the additional procedures required in the application process for the support of the APM.

6.2.3.1 Transit Node

On receipt of an APM_Transit Indication primitive, the contents are passed transparently through the transit exchange and sent to SACF in the APM_Transit Request primitive.

6.2.3.2 Public Initiating Node (PIN)

6.2.3.2.1 APM signalling association acknowledgement mechanism

No specific procedures for the PIN.

6.2.3.2.2 Segmentation co-ordination

The co-ordination of segmentation has to be performed by the receiving node. To achieve this co-ordination, the AP shall not complete the processing of information received (basic call control) in conjunction with the reception of the More_APP_Info indication primitive.

On reception of the End_APP_Info indication primitive the AP may complete the processing of the received information.

6.2.3.3 Public Addressed Node (PAN)

6.2.3.3.1 APM signalling association acknowledgement mechanism

The reception of the More_APP_Info primitive indicates that APP segmentation/reassembly procedures are underway. If the More_APP_Info indication primitive is received in conjunction with the receipt of an IAM the signalling association shall be acknowledged by sending the primitive corresponding to the appropriate first backwards message (e.g. ACM "no indication", CPG or APM) together with the APM_Acknowledgement request primitive to SACF.

6.2.3.3.2 Segmentation co-ordination

See subclause 6.2.3.1.2.

6.2.3.3.3 UCEH_Release indication

On reception of the UCEH_Release indication primitive, a release procedure is initiated with cause 79 - Service or option not implemented, unspecified. The release of an ISUP public basic call is described in EN 300 356-1[1].

6.2.4 Primitive Contents

Table 2 describes the contents of the primitives passed across interface (a).

The More_APP_Info indication primitive, the APM_Acknowledgement request primitive and the End_APP_Info indication primitives are empty.

Table 2: Contents of the UCEH_Release Ind primitive

Parameter	Mandatory/Optional
Reason	M

Table 3: Contents of the APM_Transit Ind /Req primitive

Parameter	Mandatory/Optional
Transit_Data	M

7 Single Association Control Function (SACF)

7.1 Introduction

The main objective of SACF is to receive/deliver primitives from/to the appropriate entity and to perform a distribution function where appropriate. The flow of information is from the AP (interface a) towards NI (interface f) or vice versa, therefore the SACF is also responsible to ensure that when multiple primitives are generated by the ASEs towards the AP, that they are delivered across the interface together to ensure the correct associations are maintained. The SACF described here only defines the mapping and functions related to the Application Transport Mechanism. The SACF functionality related specifically to the public basic call functionality or the APM-user application functionality is beyond the scope of the present document.

The interfaces referenced herein are illustrated in subclause 5.2, figure 2. Subclause 5.2.3 also provides examples of the "Dynamic primitive flows".

The primitives on the interface between the AP and SACF, interface (a), that are not specifically for the definition of the APM (i.e. public basic call and APM user applications are beyond the scope of the present document).

The application specific primitives on the interface between SACF and APM-user ASE, interface (b) is beyond the scope of the present document.

The primitives used on the interface between SACF and UCEH ASE, (c) are defined in subclause 10.3.

The parameters in these primitives are listed in tables 21 to 22.

The primitives used on the interface between SACF and APM ASE, (d) are defined in subclause 9.1.

The parameters in these primitives are listed in table 18.

The primitives related to ISUP basic call on the interface between SACF and ISUP ASE, interface (e) are beyond the scope of the present document. Primitives associated with the APM's use of the ISUP ASE's services are defined in subclause 8.1. Their associated contents are defined in subclause 8.3.

7.2 Outgoing Messages

On receipt of a primitive (request or response) from the AP via interface (a), the SACF issues appropriate primitive(s) to the ASEs, populating the parameters in the generated primitives from the appropriate subset of the parameters received from the AP. The SACF also performs a distribution function of the responding primitives received from the ASEs prior to sending the resulting primitive to NI via interface (f).

The APM-user ASE shall send application data to the APM ASE, via SACF, using an APM_U_Data request primitive via interface (b). Note that a number of APM-user ASEs may generate the APM_U_Data primitive. A parameter received by the APM ASE within the APM-Data primitive is identifiable through its unique relationship with an application context, and hence the appropriate APM-user ASE. Other APM related primitives on interface (b) are defined in subclause 11.1. Their associated contents are defined in subclause 11.3.

Tables 4 to 8 show the mapping of primitives performed by the SACF.

Table 4: Mapping from APM-user ASE to APM ASE primitives

Interface (b), APM-user ASE	Interface (d), APM ASE
APM_U_Data	APM_Data

Table 5: Mapping from UCEH ASE to APM ASE primitives

Interface (c), UCEH ASE	Interface (d), APM ASE
APM_U_Data	APM_Data

Table 6: Mapping from APM ASE to ISUP ASE primitives

Interface (d), APM ASE	Interface (e), ISUP ASE
APM _Transfer	APM_Transfer

Table 7: Mapping from AP to APM ASE primitives

Interface (a), AP ASE	Interface (d), APM ASE
APM_Acknowledgement	APM_Acknowledgement
APM _Transit	APM_Transit

Table 8: Mapping from ISUP ASE to NI primitives

Interface (e), ISUP ASE	Interface (f), NI
Transfer	Transfer

7.3 Incoming Messages

On receipt of a Transfer Indication primitive (see table 26) from the NI, the SACF distributes the contents of the primitive to the ISUP ASE. In the case that an APP is present, the ISUP ASE will then pass it to the APM ASE via the APM_Transfer indication primitive (see table 16). More than one APP may be present in the ISUP message, hence multiple APPs may be passed to the APM ASE. When the APM ASE is ready to pass the received information to the APM-user ASE, then it sends the APM_Data indication primitive to SACF. The SACF will distribute the primitive(s) to the appropriate APM-user ASE(s) in the APM_U_Data indication primitive according to the context identifier value which is used to uniquely identify the APM-users.

Tables 9 to 13 show the mapping of primitives performed by the SACF. For received messages, the mapping of primitives is the reverse to that described in the tables for outgoing messages in subclause 7.2.

Table 9: Mapping from UCEH ASE to APM-user ASE primitives

Interface (c), UCEH ASE	Interface (b), APM-user ASE
APM_ Error	APM_U_Error

Table 10: Mapping from UCEH ASE to AP primitives

Interface (c), UCEH ASE	Interface (a), AP
UCEH_Release	UCEH_Release

Table 11: Mapping from APM ASE to AP primitives

Interface (d), APM ASE	Interface (a), AP
APM _Transit	APM_Transit
More_APP_Info	More_APP_Info
End_APP_Info	End_APP_Info

Table 12: Mapping from APM ASE to UCEH ASE primitives

Interface (d), APM ASE	Interface (c), UCEH ASE
APM_Error	APM_UCEH_Error

Table 13: Mapping from NI to ISUP ASE primitives

Interface (f), NI	Interface (e), ISUP ASE
Transfer	Transfer

8 ISDN User Part (ISUP ASE)

The ISUP ASE is defined in EN 300 356-1 [1]. Its main responsibilities are basic call procedures and the handling of protocol errors and unrecognized information handling. The monolithic nature of the present document means that both Public Call Control and Protocol Control functionality are defined together. It is not the intention of the present document to re-define EN 300 356-1 [1] in ALS format, therefore they are referenced en-bloc as the ISUP ASE. Conceptually, this should be considered to represent a logical division between the protocol control functionality within the ISUP ASE and its associated call control functionality within the application process. The modelling and interfaces with respect to this are beyond the scope of the present document.

8.1 Primitive interface

The SACF uses the services provided by the ISUP ASE primitive interface (see interface (e) in figure 2) as listed in table 14.

Table 14: Primitives between ISUP ASE and ISUP SACF

Primitive name	Types
APM_Transfer	Indication/Request

ISUP ASE uses the SACF service primitives: Transfer request/indication.

It is also a receiver of the indications: Remote_Status, Destination_Available, Destination_Unavailable.

8.2 Procedures

The procedures of the ISUP ASE are defined in reference [14]. With respect to the APM; on reception of APM_Transfer request primitive(s) containing APM-user information, it is the responsibility of the ISUP ASE to include the APP(s) within a suitable call control message, when available, or alternatively to generate an Application Transport Mechanism message (APM message; see EN 300 356-1 [1]). Similarly it shall handle the reception of the APP parameter(s) and the transfer of the received data in the APM_Transfer indication primitive(s). The APM message can only be sent/received on a non-idle circuit and does not cause a state change. The ISUP parameter compatibility instruction indicators should have the same settings as the ATII. In the case where a message contains more than one APP, the ATII setting will be according to the needs of each APM-user application; however the ISUP parameter compatibility settings shall be according to the most stringent case with regard to the APM-users.

8.3 Primitive contents

Table 15: Contents of the APM_Transfer Ind/Req primitive

Parameter	Mandatory/Optional	Reference
Application Transport (APP)	M	EN 300 356-1 [1]

9 Application Transport Mechanism (APM) ASE

The APM ASE specifies the extensions to ISUP in order to provide the underlying services to the APM-user ASE(s) and support the application's (APM-user's) information flows. The APM ASE is responsible for the signalling procedures and the parameters specific to the APM.

The APM ASE provides a segmentation mechanism to the APM-user(s) such that the ISUP message length limitation does not restrict the APM-user information flows. The maximum supported size of APM-user information is 2 048 octets in length, per APM-user. The APM ASE may serve many APM-users which are differentiated by their Context Identifiers.

9.1 Primitive interface

The ISUP SACF uses the services provided by the APM ASE primitive interface (see interface (d) in figure 2) as listed in table 16.

Table 16: Primitives between APM ASE and ISUP SACF

Primitive name	Types
More_APP_Info	Indication
End_APP_Info	Indication
APM_Transit	Indication/Request
APM_Acknowledgement	Request
APM_UCEH_Error	Indication
APM_Data	Indication/Request

The APM ASE uses the services provided by the ISUP SACF primitive interface (see interface (d) in figure 2) as listed in table 17.

Table 17: Primitives between ISUP SACF and APM ASE

Primitive name	Types
APM_Transfer	Indication/Request

9.2 Procedures

9.2.1 Normal procedures - sending

On receipt of an APM_Data request primitive, the APM ASE constructs the APP according to the definition in EN 300 356-1 [1]. The context identifier is set based on the APM-user. The Application Transport Instruction indicators are supplied by the APM-user. The segmentation indicator is set to zero (0) unless segmentation procedures apply (see subclause 9.2.4). How this is determined is implementation specific. The Application Data received from the APM-user is encapsulated and the whole APP is transferred in the APM_Transfer request primitive.

The APP is transferred in the APM_Transfer request primitive.

On receipt of an APM_Transit request primitive the APM ASE transfers the received information transparently in an APM_Transfer request primitive.

9.2.2 Normal procedures - receiving

On receipt of an APM_Transfer indication primitive, the subsequent action depends on whether it was received at call setup or at another time.

9.2.2.1 Procedures at call setup

On receipt of an APM_Transfer indication at call setup, the action depends on whether this is the addressed node for this call. If not, the APM ASE notes that this node is a "pass-on" node for this context and transfers the APP parameter unchanged in an APM_Transit indication primitive.

If this is the addressed node, the action depends on whether the context in question is supported at this node:

- If the context is supported, the parameter is checked to see if segmentation has occurred. If so the segmentation reassembly procedure applies (see subclause 9.2.4) and the completely reassembled Application Information is sent in the APM_Data indication primitive.

If the context is not supported, then an APM_UCEH_Error indication primitive is sent. The Application Context and ATII are set as received in the APM_Transfer primitive and the Reason is set to "Unidentified Context". If the parameter had been segmented the APM ASE notes that the reassembly procedure for this context has been abandoned.

9.2.2.2 Procedures at other times

On receipt of an APM_Transfer indication, the APM ASE checks whether this is a "pass-on" node for this context. If so, the APP parameter is passed on unchanged within an APM_Transit indication primitive.

If not, the APM ASE checks whether it is expecting an acknowledgement to an initial segment sent at call setup according to the segmentation procedures of subclause 9.2.4. If so, the segmentation procedures apply. If the APM ASE is not expecting such an acknowledgement, the action depends on whether the context in question is supported at this node:

- If the context is supported, the parameter is checked to see if segmentation has occurred. If so, the segmentation reassembly procedure applies (see subclause 9.2.4) and the completely reassembled Application Information is sent in the APM_Data indication primitive.

- If the context is not supported, and this is a transit node, then the APM ASE notes that this node is a "pass-on" node for this context and transfers the APP parameter unchanged in an APM_Transit indication primitive.

If the context is not supported and this is not a transit node, then if this is the first or only segment of a sequence, an APM_UCEH_Error indication primitive is sent. The Application Context and Application Transport Instruction Indicators are set as received in the APM_Transfer primitive and the Reason is set to "Unidentified Context". If the parameter had been segmented the APM ASE notes that the reassembly procedure for this context has been abandoned. Subsequent segments in this case are discarded.

9.2.3 Sending of acknowledgement

On receipt of an APP_Send request primitive, an APP will be constructed with a context identifier of an APP currently under reassembly. The ATII will be set to "release call" and "do not send notification" and the Application Information field shall be empty. This APP will then be sent in an APM_Transfer request primitive.

9.2.4 Segmentation

Segmentation procedures may be independently applied to several APM-user applications simultaneously. The procedures in subclause 9.2.4.1 describe the handling of segmentation and reassembly for application data regarding a given APM-user (context identifier). It should be noted that when multiple APM-users are utilizing the services of the APM for a single call, the procedures may be applied concurrently and independently with respect to each APM-user.

The initial segment for each context shall be transported in the first message and this initial segment size may be of zero length. The first segment may be an ISUP call control message (when available) and this procedure ensures that the reassembling node is able to associate the APM information with the call control message. A maximum of one call control message can be associated with the transport of segmented application data. Any additional segments will be included in the APP within subsequent APM message(s). In order to associate all the reassembled application data with the call control message the AP is informed when reassembly procedures are invoked by sending the More_APP_Info indication primitive. On completion of reassembly for all contexts for which initial segments were received in the first message, the End_APP_Info indication primitive is sent.

There would be a possibility that user information segments could be lost when sent forward during call setup if sent before the receipt of a first backwards message. To avoid this situation occurring, the PIN shall not send subsequent segments to an IAM message until a first backward message is received containing an APP, which implicitly indicates that a path has been successfully routed through the network to the PAN. The PAN initiates the backwards acknowledgement indication only in the case that an IAM is received containing an APP which indicates that segmentation has occurred.

9.2.4.1 Procedures for segmentation

The following rules apply when Application Information in an APM_Data request primitive has to be segmented:

a) The maximum number of segments is 10 (one initial, possibly empty, segment plus up to 9 segments). The maximum size of Application Information that can be supported is 2 048 octets. If the Application Information is too long to be segmented, then the local maintenance function shall be notified, and the APM_Data request primitive discarded.

b) The Encapsulated Application Information field in the first segment shall begin with the first octet of the application Information and sequentially fill the Encapsulated Application Information field. (Alternatively the first segment may contain zero octets of Application Information, and the second segment is filled as described.) The Segmentation indicator field will be set to indicate "new sequence", the Number of Segments Remaining field to indicate the number of segments that remain to be sent and a Segmentation Local Reference value included that is unique to the call.

c) The encapsulated information field in each subsequent segment shall begin with the first octet following the last octet transmitted in the previous segment, thus continuing segmentation until no further encapsulated application data octets remain. The Segmentation indicator field shall indicate "subsequent segment to first segment" and the Number of Segments Remaining field shall be decremented to indicate the number of segments that remain to be sent and the segmentation local reference shall have the same value as sent in the first segment.

d) Once the first segment has been transmitted, then all remaining segments of that Application Information shall be sent except in the case that the first segment is sent in an IAM in which case the reception of an APP (empty) shall be awaited prior to the sending of subsequent segments.

e) Only failure conditions shall cause the transmission of a segmented message to be aborted. In this case the maintenance function is to be notified.

f) The ATII should be set as provided by the APM-user in the APM-Data request primitive for the first and all subsequent segments.

9.2.4.2 Procedures for reassembly

The following rules apply to the receipt and reassembly of segmented Application Information:

a) The reassembly function, on receiving an APM_Transfer indication primitive containing an APP with the Sequence indicator field indicating "new sequence" and the Number of Segments Remaining field set to a value greater than zero (0), and not exceeding nine (9), shall treat that Encapsulated Application Information as a valid first segment and save the segment contents, and the received Segmentation Local Reference value. (The first segment may have zero octets of Encapsulated Application Information.) Timer T-reass shall be started on receipt of a valid first segment.

b) The reassembly function, on receiving a subsequent segment with the Sequence Indicator field indicating "subsequent segment to first segment" and the number of segments remaining field set to a value one less than the value in the previously received segment and the same Segmentation Local Reference value as received in the first segment shall treat the segment as a valid next segment.

c) A reassembly function, on receiving a valid next segment in which the number of segments remaining field has a value greater than zero, shall save the segment contents along with the saved contents of the previous segment or segments.

d) A reassembly function, on receiving a valid next segment in which the number of segments remaining field has a value equal to zero, final segment, shall deliver the accumulated segments, including the last segment, as the complete Application Information to the APM-user with the APM_Data indication primitive. Timer T-reass shall be stopped on receipt of the final segment of the sequence.

e) If no reassembly process is active, a reassembly function shall discard any received segment which is not a valid first segment and send the APM_UCEH_Error Indication primitive indicating "reassembly error".

This applies to:

- Segments with a Sequence Indicator coded "Subsequent segment to first segment".
- Segments in which the Number of Segments Remaining field has a value exceeding 9.

f) A reassembly function shall discard the received and any saved segments on receipt of any subsequent segment which is not a valid next segment and send the APM_UCEH_Error Indication primitive indicating "reassembly error".

This applies to:

- Segments with a Number of Segments Remaining field value that is not decremented from the value of the previous segment.
- Different Segmentation Local Reference from that received in the first segment.

g) A reassembly function shall discard any saved segments on receipt of any subsequent segment which is not a valid next segment and send the APM_UCEH_Error Indication primitive indicating "reassembly error".

This applies to:

- Segments with a Sequence Indicator coded "New sequence".

h) On expiry of timer T-reass, the reassembly function shall discard all segments received for the current sequence and send the APM_UCEH_Error indication primitive indicating "reassembly error".

9.3 Primitive Contents

The following primitives are already defined elsewhere:

APM_Transfer primitive is defined in table 15.

More_APP_Info primitive is defined in subclause 6.2.4.

End_APP_Info primitive is defined in subclause 6.2.4.

APM_Acknowledgement primitive is defined in subclause 6.2.4.

APM_Transit primitive is defined in table 3.

APM_UCEH_Error primitive is defined in table 22.

Table 18 defines the primitive used across the SACF/APM ASE interface (d).

Table 18: Contents of the APM_Data Ind/Req primitive

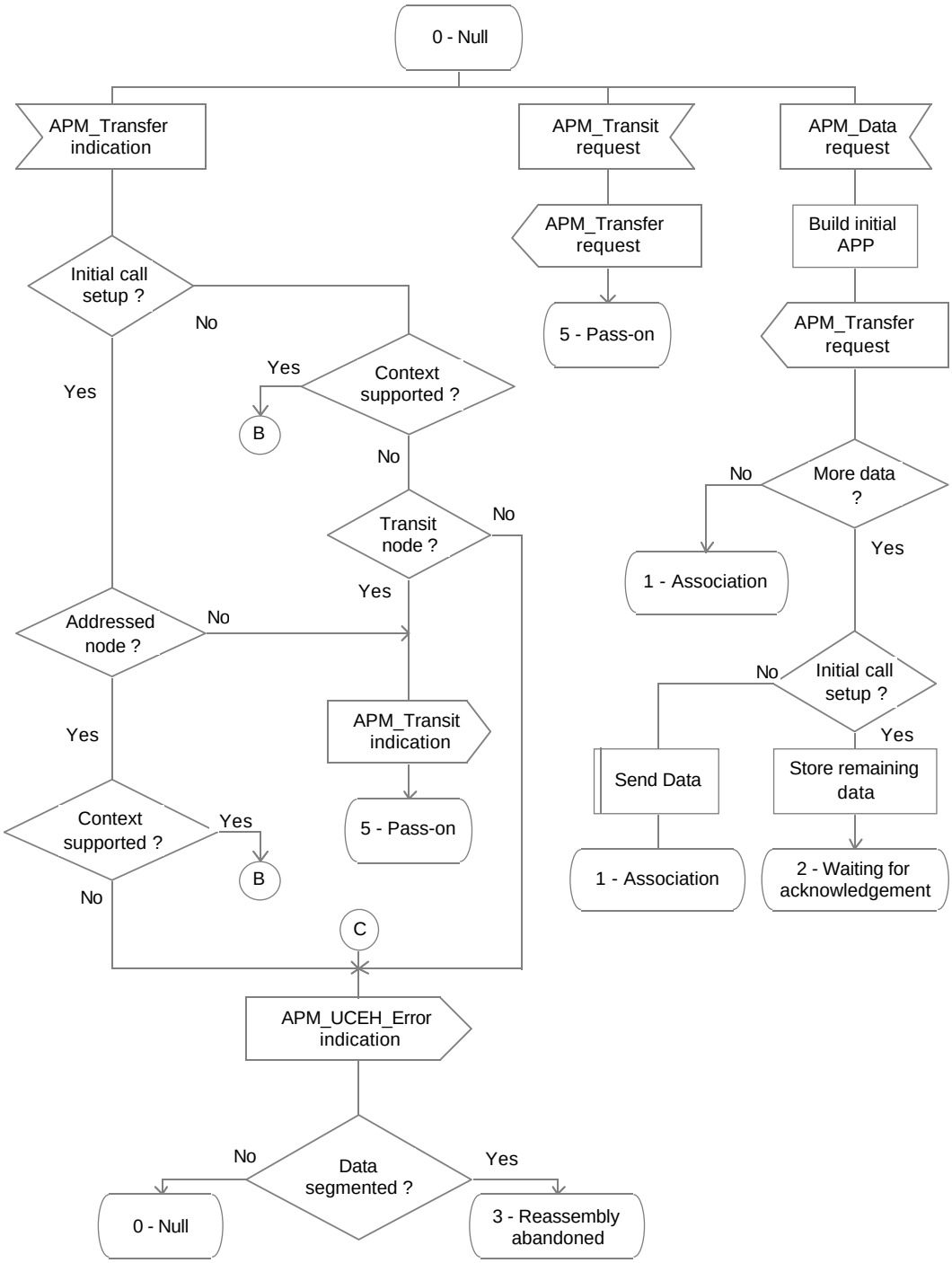
Parameter	Mandatory/Optional
Application Context Identifier	M
Application Transport Instruction Indicators	M
Application Information	M

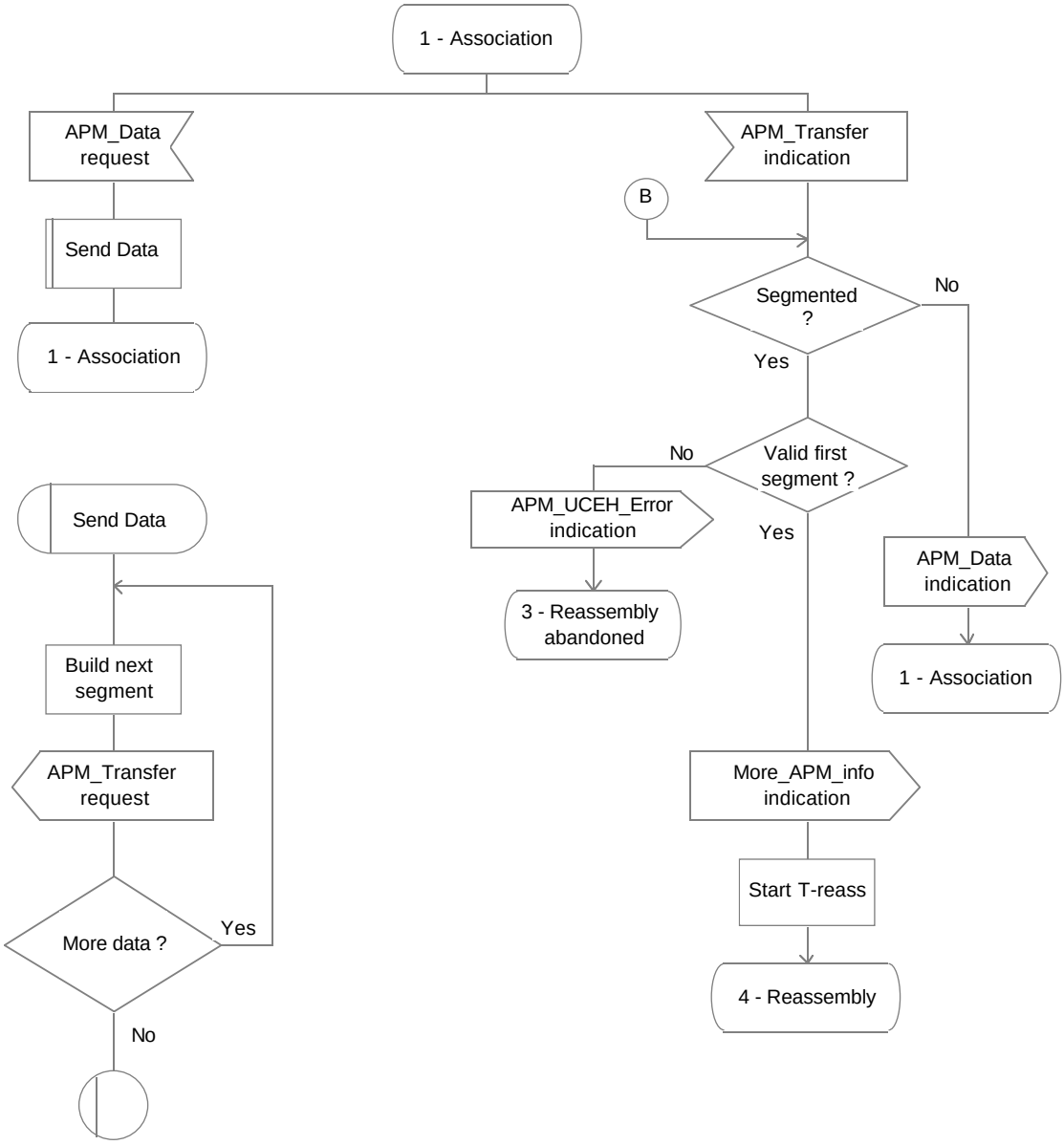
9.4 SDL Diagrams

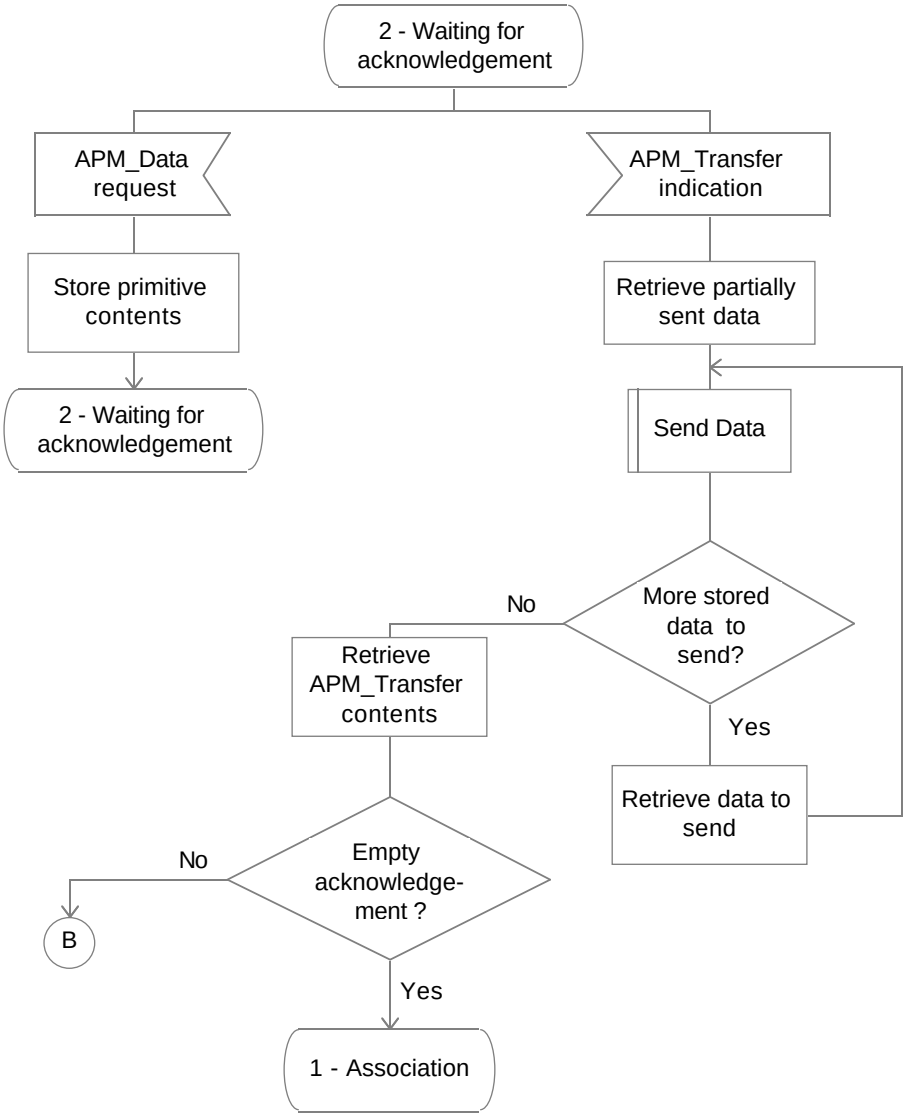
Working Document - SDLs for the APM
revision 1

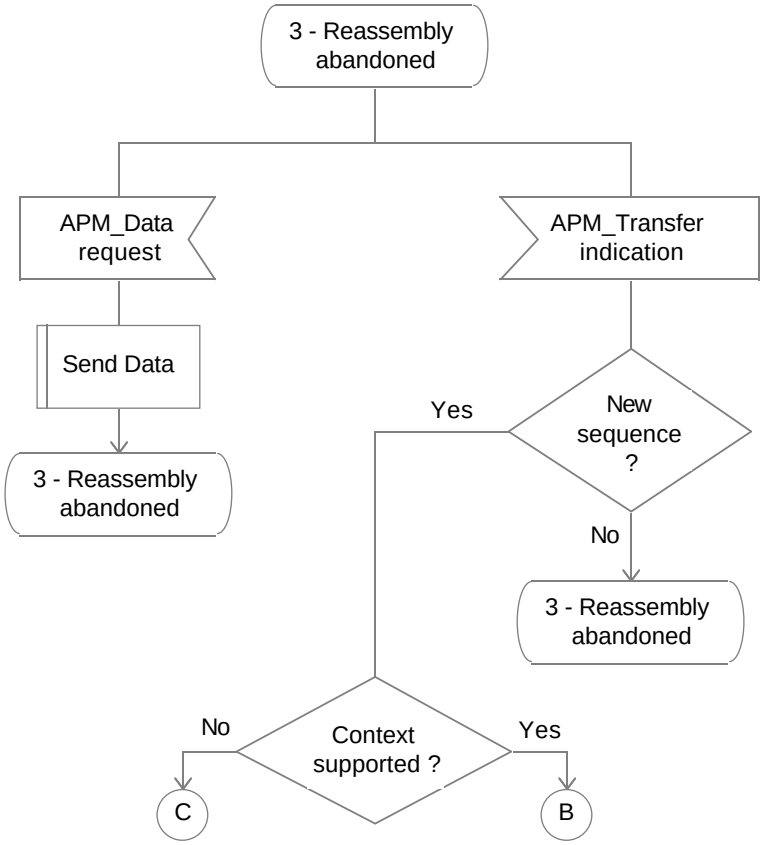
These SDLs describe a state machine which receives primitives corresponding to a single context. When a primitive arrives at the APM ASE for a context previously unused on the call, a new instance of the SDL is created starting in state '0 - Null'. Subsequent primitives for the same context on this call are passed to this instance.

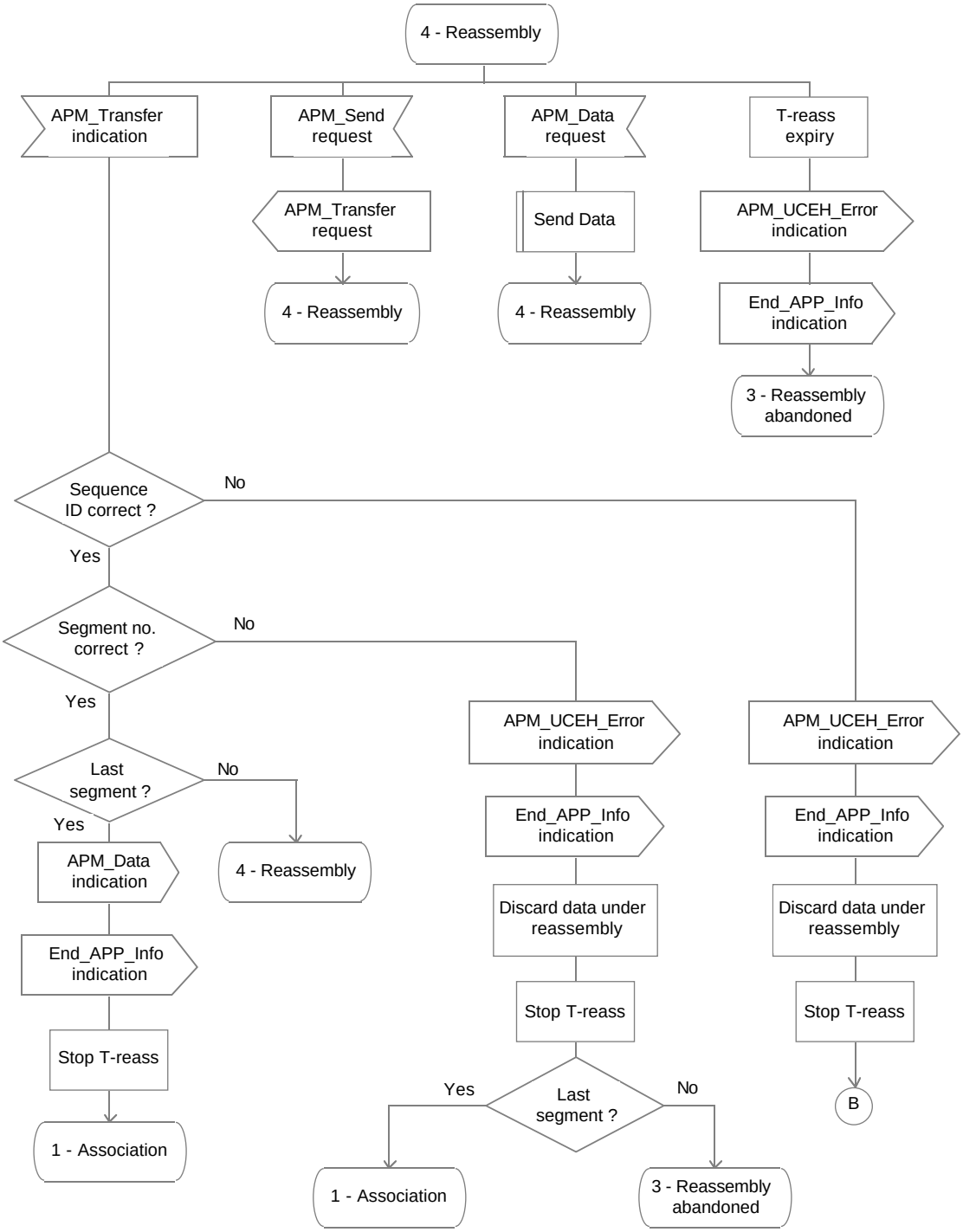
It is assumed that the APM ASE has access to information as to which contexts are recognised, the type of the node, transit or end node, and whether this is the addressed node.

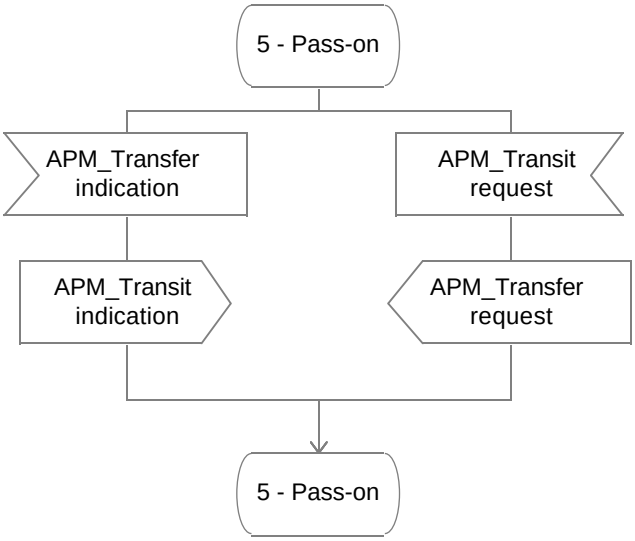



1 - Association
APM_Data request
Send Data
1 - Association
APM_Transfer indication
B
Segmented ?
No
Yes
Valid first segment ?
No
Yes
APM_UCEH_Error indication
3 - Reassembly abandoned
APM_Data indication
1 - Association
More_APM_info indication
Start T-reass
4 - Reassembly
Send Data
Build next segment
APM_Transfer request
More data ?
Yes
No


2 - Waiting for acknowledgement
APM_Data request
APM_Transfer indication
Store primitive contents
Retrieve partially sent data
2 - Waiting for acknowledgement
Send Data
More stored data to send?
No
Yes
Retrieve APM_Transfer contents
Retrieve data to send
Empty acknowledge-ment ?
No
Yes
B
1 - Association


3 - Reassembly abandoned
APM_Data request
APM_Transfer indication
Send Data
3 - Reassembly abandoned
New sequence ?
Yes
No
3 - Reassembly abandoned
No
Context supported ?
Yes
C
B


4 - Reassembly
APM_Transfer indication
APM_Send request
APM_Data request
T-reass expiry
APM_Transfer request
Send Data
APM_UCEH_Error indication
4 - Reassembly
4 - Reassembly
End_APP_Info indication
3 - Reassembly abandoned
Sequence ID correct ?
No
Yes
Segment no. correct ?
No
Yes
Last segment ?
No
Yes
APM_Data indication
4 - Reassembly
End_APP_Info indication
Stop T-reass
1 - Association
APM_UCEH_Error indication
End_APP_Info indication
Discard data under reassembly
Stop T-reass
Yes
Last segment ?
No
1 - Association
3 - Reassembly abandoned
APM_UCEH_Error indication
End_APP_Info indication
Discard data under reassembly
Stop T-reass
B


5 - Pass-on
APM_Transfer indication
APM_Transit request
APM_Transit indication
APM_Transfer request
5 - Pass-on

10 Unidentified Context and Error Handling (UCEH) ASE

The UCEH ASE handles error cases such as unidentified context values and reassembly errors. The UCEH ASE takes action according to the ATII received together with the application information in the APP. Based on the Release Call Indicator, the call may be released or allowed to proceed. The Send Notification Indicator determines if a notification is sent. On reception of such a notification at the node which sent the original application information, the UCEH ASE is responsible for informing the appropriate APM-user application that the error has occurred. For example, to notify the application located at the PIN that the desired context identifier value was not supported at the PAN. The UCEH ASE is a mandatory ASE in the AEI that supports the APM ASE.

10.1 Introduction of unidentified context handling mechanism

The unidentified context handling mechanism describes the case when an exchange (PIN), supporting an application identified by a specific context identifier, initiates a signalling association with a remote exchange (PAN) which has the APM capability but does not support the application indicated by the context identifier. The unidentified context handling mechanism ensures this situation is handled according to the specific needs of the initiating application.

10.1.1 Unidentified context handling (PAN)

When a PAN receives an unidentified context identifier value the UCEH ASE takes action according to the ATII. Based on the Release Call Indicator, the call may be released or allowed to proceed. The Send Notification Indicator determines if a notification is sent back from the PAN towards the PIN.

Figure 9 illustrates the dynamic flow if information from the point where the APM ASE identifies that the APM-user is not supported. In the case that there is no APM-user supporting the context identifier of a received APP, then the APM ASE sends an APM_UCEH_Error indication primitive indicating the context that is not supported and supplies the received ATII. This initiates the unidentified context handling procedures. These procedures can be seen as a user of the APM mechanism by the manner in which the notification is transported.

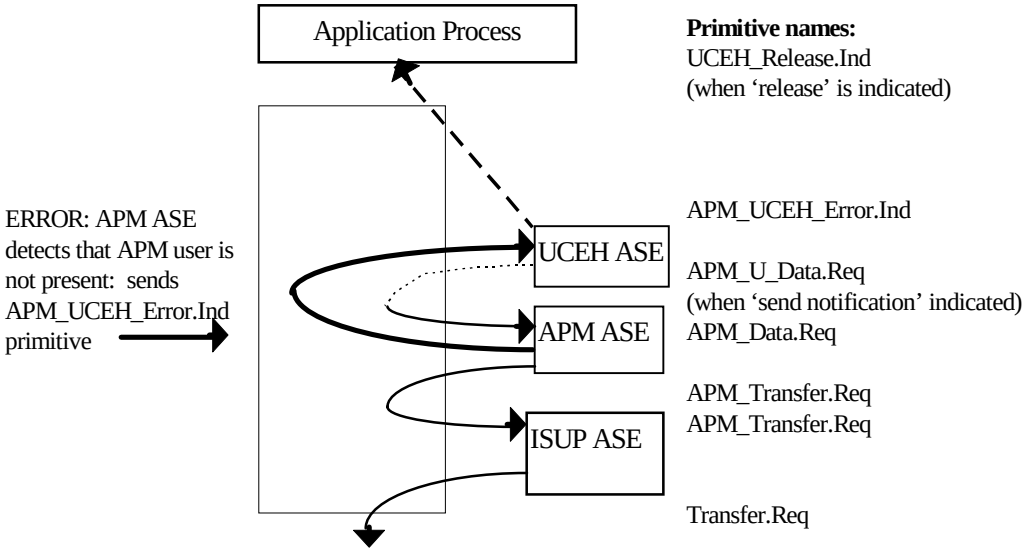


Figure 9: Dynamic flow in PAN when APM-user not present

10.1.2 Unidentified Context Handling (PIN)

When an APP is received with Context Identifier "Unidentified Context and Error Handling ASE", the contents are passed to the UCEH ASE in the same manner as for any other APM-user. The Context Identifier carried by the Application Transport Notification Information contained within the Encapsulated application information field of the APP will be used to identify the APM-user to which the APM_Error indication primitive will be sent from the UCEH ASE. The APM_Error indication primitive will indicate that the reason for the error was that the peer APM-user was not present at the PAN.

Figure 10 illustrates the dynamic flow of information for this case in the PIN.

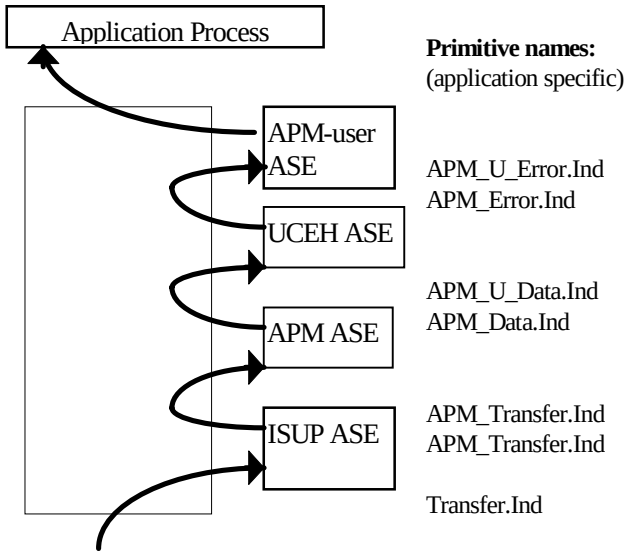


Figure 10: Dynamic flow in PIN to handle reception of APM notification.

10.2 Reassembly error handling

When a reassembly error occurs in the APM ASE, the APM_UCEH_Error primitive is sent, via SACF, to the UCEH ASE indicating "reassembly error". Actions taken by the UCEH ASE are according to the ATII. If a notification is to be sent, the reason given indicates "reassembly error". A reassembly error may occur for any segmented message in either a PIN or a PAN, therefore the procedures are equally applicable to both.

10.3 Primitive interface

The SACF use the services provided by the UCEH ASE primitive interface (see interface (c) in figure 2) as listed in table 19.

Table 19: Primitives between UCEH ASE and SACF

Primitive name	Types
APM_Error	Indication
UCEH_Release	Indication

The UCEH ASE use the services provided by the SACF primitive interface (see interface (c) in figure 2) as listed in table 20.

Table 20: Primitives between SACF and UCEH ASE

Primitive name	Types
APM_UCEH_Error	Indication
APM_U_Data	Indication/Request

10.4 Procedures

10.4.1 Normal procedures - remote error handling

When the UCEH ASE receives an APM_U_Data indication primitive, the Context Identifier within the received Notification parameter (clause 13) is used to determine the APM-user that should be notified. An APM_Error indication primitive is sent to the APM-user via SACF indicating the Reason for the error. The maintenance function shall be notified.

10.4.2 Normal procedures - local error handling

When an APM_UCEH_Error indication primitive is received by the UCEH ASE, the ATII are checked to determine the appropriate actions. If "send notification" is set then the Notification (clause 13) parameter shall be constructed and shall indicate the APM-user Context Identifier, the reason and the ATII be set to "release call" and "do not send notification".

This shall then be sent to the APM ASE via the APM_U_Data request primitive for transport. If "release call" is set then a UCEH_Release indication primitive is sent to the AP via SACF. The maintenance function shall be notified.

10.4.3 Exceptional procedures - context identifier error

If a context identifier is received as "no information" then the information is discarded and the maintenance function will be notified.

10.5 Primitive contents

Tables 21 to 22 define the primitives used across the SACF/UCEH ASE interface (interface (c) in figure 2). The APM_U_Data primitive is described in table 24. The UCEH_Release primitive is described in table 2.

Table 21: Contents of the APM_Error Ind primitive

Parameter	Mandatory/Optional
Notification	M

Table 22: Contents of the APM_UCEH_Error Ind primitive

Parameter	Mandatory/Optional
Reason	M
Application Context Identifier	M
Application Transport Instruction Indicators	M

11 Application Transport Mechanism user (APM-user) ASE

The APM-user ASE is responsible for the detailed protocol control signalling aspects specific to the application it represents. The definition of the application is beyond the scope of the present document. The APM-user ASE utilizes a specific interface towards SACF in order to access the services of the APM ASE.

An APM-user is uniquely identified by the SACF through its "context identifier value". This represents the application that it supports and defines the "context" in which this APM-user operates.

11.1 Primitive interface

The SACF/APM-user ASE interface (interface (b) in figure 2) is application specific and therefore beyond the scope of the present document, except for the primitive interface part used to access the services of the APM ASE. The APM specific primitives are listed in table 23.

Table 23: Primitives between APM-user ASE and SACF

Primitive name	Types
APM_U_Data	Indication/Request
APM_U_Error	Indication

APM-user ASE may be a receiver of the SACF service primitive indications: Remote_Status, Destination_Available, Destination_Unavailable dependant on the needs of the given application.

11.2 Procedures

11.2.1 General

The procedures for the APM-user ASE are application specific and therefore beyond the scope of the present document.

11.2.2 Signalling congestion

The following mechanism for the handling of signalling congestion is provided as a guideline for all applications that generate No.7 signalling load. APM-users should consider the inclusion of the following signalling congestion handling procedure in the context of the application:

In order to avoid signalling link congestion in the No.7 network, it is necessary that applications that contribute signalling load towards a congested link limit such signalling in a controlled manner. On receipt of a congestion indication (the result of the MTP generating an MTP_status indication primitives (see EN 300 356-1 [1])) the application should reduce signalling traffic load in several steps. The application may receive such an indication directly from MTP in the form of the MTP_status indication primitive or it may be received indirectly from MTP in alternative primitives when the application is not a direct user of MTP services. For a direct user of MTP services, as is the case for ISUP and applications that make use of signalling association established by these protocols (e.g. APM-user applications) the following procedure is applicable on a destination basis. In the case where the application is not a direct user of the MTP services and that the notification that it receives does not allow for knowledge of signalling destination point codes, as is the case for TC-user applications, then the procedure will apply to the instance to which the received primitive applies.

When the first congestion indication is received by the application, the signalling traffic load is reduced by one step. At the same time two timers T-C1 and T-C2 are started. During T-C1 all received congestion indications are ignored in order not to reduce traffic too rapidly. Reception of a congestion indication after the expiry of T-C1, but still during T-C2, will decrease the signalling load by one more step and restart T-C1 and T-C2. This step wise reduction of the application signalling load is continued until maximum reduction is obtained by arriving at the last step. If T-C2 expires (i.e. no congestion indications having been received during the T-C2 period) signalling load will be increased by one step and T-C2 will be restarted unless full signalling load has been resumed.

Timers T-C1 and T-C2 have the following recommended range:

T-C1 = 300 ms to 600 ms;

T-C2 = 5 s to 10 s.

The number of steps of signalling reduction and the type and/or amount of increase/decrease of signalling load at the various steps are considered to be an application matter and should be selected so as to reduce and increase the signalling in a smooth manner.

The two timer method, as described above, would allow for a smooth reduction and resumption of signalling towards the MTP in the case of congestion.

The use of different timer values for different applications would allow for a mechanism of prioritization of the applications. These timer values should be definable by the operator.

11.3 Primitive contents

Table 24: Contents of the APM_U_Data Ind/Req primitive

Parameter	Mandatory/Optional
Application Context Identifier	M
Application Transport Instruction Indicators	M
Application Data	M

Table 25: Contents of the APM_U_Error Ind primitive

Parameter	Mandatory/Optional
Notification	M

12 Network Interface function

12.1 Introduction

The Network Interface (NI) function is the part of ISUP that provides a transport interface for instances of the ISUP AEI. The MTP-3 Service Primitive interface is a single interface within an exchange, but multiple instances of signalling associations exist within one exchange. NI associates the signalling instance representation (session layer, ITU-T Recommendation Q.1400 [2]) of the external interface to the internal instances. NI also represents the presentation layer (ITU-T Recommendation Q.1400 [2]) for ISUP.

12.2 Primitive interfaces

12.2.1 MTP interface

The interface (see interface (g) in figure 2) to the MTP-3 is the Service Primitive interface defined in EN 300 356-1 [1].

12.2.2 AEI interface

The primitives listed in table 26 are present on the interface between the NI and the SACF in the ISUP AE (see interface (f) in figure 2).

Table 26: Primitive between NI and SACF

Primitive name	Types
Transfer	Request/Indication
Remote_Status	Indication
Destination_Unavailable	Indication
Destination_Available	Indication

12.3 Procedures

12.3.1 Formats and codes

The NI performs the presentation layer (ITU-T Recommendation Q.1400 [2]) function for the ISUP. The details of ISUP formats and codes can be found in EN 300 356-1 [1].

MTP_Transfer indication primitives received from the MTP (see interface (g) in figure 2) are decoded according to the formatting rules and the procedures defined in EN 300 356-1 [1].

12.3.2 Distribution

MTP_Transfer indication primitives received from the MTP are distributed to AEIs. Before this distribution can be performed, the circuit to which the incoming message is addressed needs to be tied to an AEI. Through the use of the CIC, NI, OPC and DPC values of the circuit, AEI identity, mapping is performed according to the following rules:

- If the circuit value corresponds to an existing ISUP AEI the message is distributed to that AEI.
- If the circuit value does not correspond to an existing ISUP AEI, a new instance of ISUP AEI is created.

When a Transfer request primitive is received from a ISUP AEI, it is mapped into a MTP_Transfer request primitive. The AEI identity with which the message is associated is used to identify the CIC, NI, OPC and DPC values to use when sending the message. The SIO and SLS fields are completed as specified in EN 300 356-1 [1].

12.3.3 Primitive mapping

Tables 27 and 28 show the primitive mapping performed by the NI.

Table 27: NI Primitive Mappings from MTP to SACF

Interface (g) from MTP	Interface (f) to SACF
MTP_Transfer indication	Transfer Indication
MTP_Status indication	Remote_Status Indication
MTP_Pause indication	Destination_Unavailable Indication
MTP_Resume indication	Destination_Available Indication

Table 28: NI Primitive Mappings from SACF to MTP

Interface (f) from SACF	Interface (g) to MTP
Transfer request	MTP_Transfer request

13 Application transport notification information

This information is carried within the Encapsulated Information field of the APP.

	8	7	6	5	4	3	2	1
1	Ext	APM-user Context Identifier						
2	Ext	Reason						

Figure 11: Application transport notification information

a) Extension indicators

0 further octet exists

1 last octet

b) *APM-user Context Identifier*

0 No Information

1-127 Refer to "Application Context Identifier" field in the Application Transport Parameter (EN 300 356-1 [1])

c) *Reason*

0 No Information

1 Unidentified Context

2 Reassembly Error

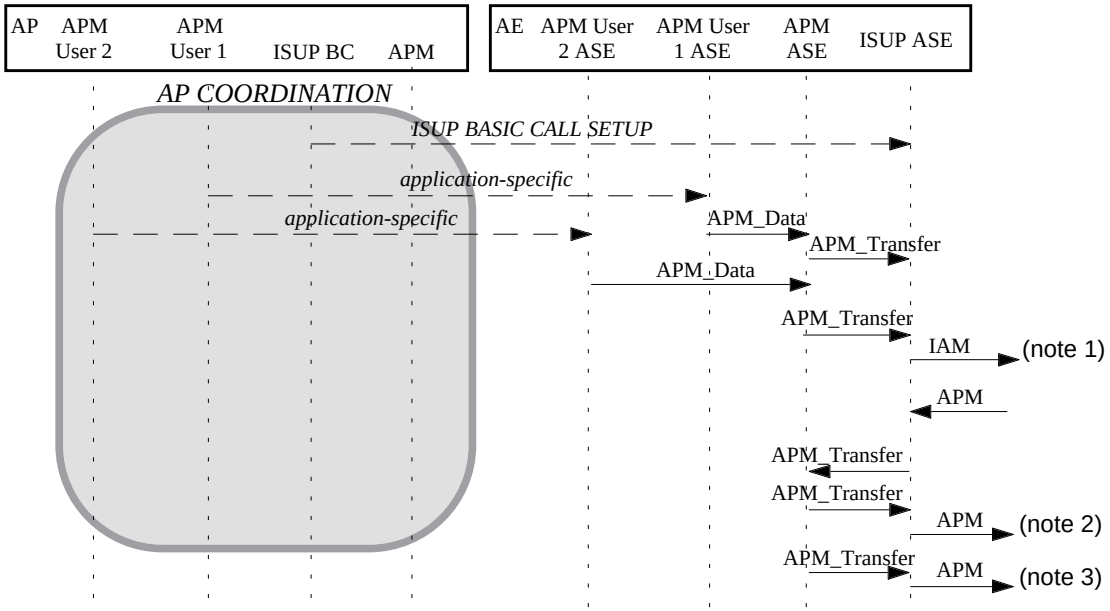
NOTE: The Application Transport Notification Parameter will be transported within the Encapsulated Application Information field.

14 Timers

Table 29: Timers

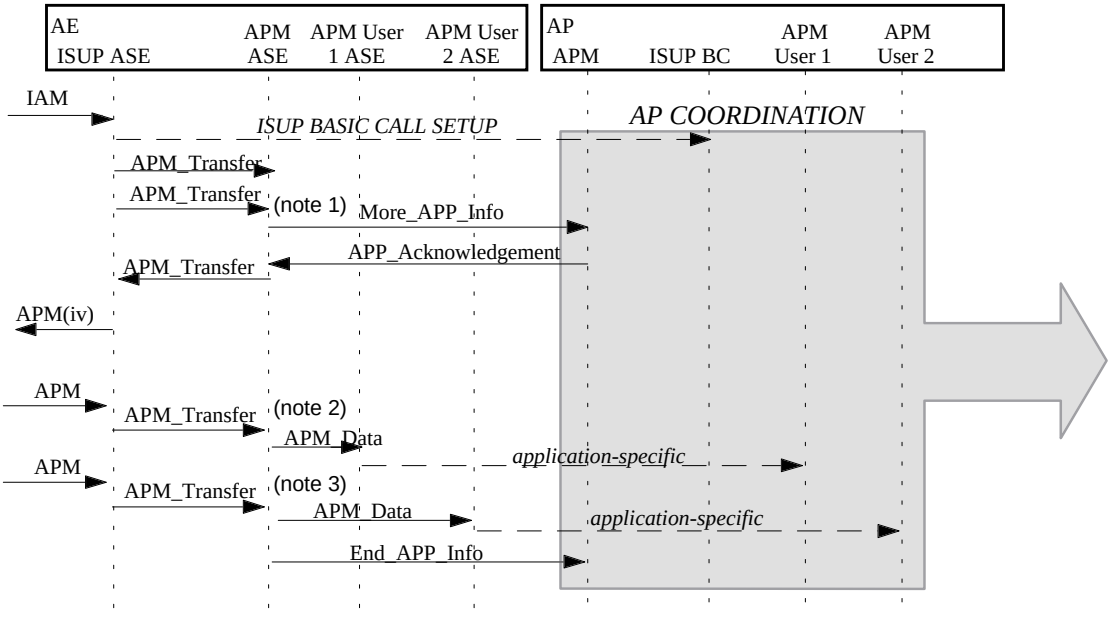
Symbol	Time-out value	Cause for initiation	Normal termination	At expiry	Reference
T-C1	300 - 600 ms	Reception of "signalling congestion" indication	Expiry of T-C1	Act upon subsequent "signalling congestion" indications whilst T-C2 is still running	11.2.2
T-C2	5 - 10 s	Reception of "signalling congestion" indication and on expiry of T-C2 unless full signalling load has been reached	Expiry of T-C2 (when full signalling load has been reached)	Increase signalling load by one step. Restart T-C2 if not carrying full signalling load	11.2.2
T-reass	10-18 s	receipt of APP indicating "new sequence" for reassembly	receipt of last segment	apply "reassembly error handling"	9.2.4.2

Annex A (informative): Example sequence diagrams of APM segmentation



NOTE 1: IAM includes initial APP for each APM User, these may be "empty".
NOTE 2: APM includes last APP segment for APM User 1.
NOTE 3: APM includes last APP segment for APM User 2.

Figure A.1: sending of an IAM containing initial APM segments for 2 Applications



NOTE 1: Contains initial APPs for each APM User.
NOTE 2: Contains last segment for APM User 1.
NOTE 3: An APM message is used if there is no concurrent call control message being sent.

Figure A.2: receipt of an IAM containing initial APM segments for 2 Applications

History

Document history				
V1.1.1	August 1997	Public Enquiry	PE 9748:	1997-08-01 to 1997-11-28