Standard ECMA-254

December 1996

ECMA

Standardizing Information and Communication Systems

Broadband Private Integrated Services Network (B-PISN) - Inter-Exchange Signalling Protocol - Generic Functional Protocol

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:** http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

Standard ECMA-254
December 1996

ECMA

Standardizing Information and Communication Systems

Broadband Private Integrated Services Network (B-PISN) - Inter-Exchange Signalling Protocol - Generic Functional Protocol

(B-QSIG-GF)

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:** http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

MB - Ecma-254.doc - 30.07.98 10:37

Brief History

This Standard is one of a series of ECMA standards defining services and signalling protocols applicable to Broadband Private Integrated Services Networks. The series uses the B-ISDN concepts as developed by ITU-T (formerly CCITT) and is also within the framework of standards for open systems interconnection as defined by ISO.

This Standard specifies the signalling protocol for use at the Q reference point in support of the Generic Functional Protocol.

The Standard is based upon the practical experience of ECMA member companies and the results of their active and continuous participation in the work of ISO/IEC JTC1, ITU-T, ETSI and other international and national standardisation bodies. It represents a pragmatic and widely based consensus.

This ECMA Standard is contributed to ISO/IEC JTC1 under the terms of the fast-track procedure for adoption as an ISO/IEC International Standard.

This Standard has been adopted by the ECMA General Assembly of December 1996.

List of corrected errata for ECMA-254

9 July 1998

Summary

Following is a summary of the errors detected and corrected in Standard ECMA-254, Broadband Services Network (B-PISN) – Inter-Exchange Signalling Protocol – Generic Functional Protocol.

References

All references to ITU-T Recommendation Q.2932.1 are faulty. The first part of each reference "1." shall be removed (global change).

e.g.

Section 4 Definitions

Corrected:
"Definitions described in ITU-T Rec. Q.2932.1 clause 3 ..."

Original:
"Definitions described in ITU-T Rec. Q.2932.1 clause 1.3 ..."

Section 3 references

Following references shall be added:

Corrected:

ITU-T Rec. E.164	Numbering plan for the ISDN era (1991)
ITU-T Rec. X.213	Information technology - Network service definition for open systems interconnection (1992)

Section 10.1.3.1 Action in the Null state

Section 10.1.3.1 of ECMA-254 refers to section 1.9.1.3.1 (should now be 9.1.3.1) of Q.2932.1.

9.1.3.1 of Q.2932.1 states that for a CO-BI SETUP message Calling and Called party number information element may be included while section 9.1.3.2 of ECMA-254 states that these two information elements are mandatory. An additional dashed item is added to section 10.1.3.1 of ECMA-254:

Corrected:

– The preceding side shall include the Called party number and the Calling party number information elements identifying the destination and the source respectively of the bearer independent signalling connection.

Section 10.1.3.9 Transport of APDUs associated with a connection-oriented bearer-independent signalling connection

The referenced section number of Q.2932.1 should be corrected.

Corrected:
… described in ITU-T Rec. Q.2932.1 clause 9.1.3.9 shall apply.

Original:
… described in ITU-T Rec. Q.2932.1 clause 1.91.3.9 shall apply.

ASN.1 Errors

Following syntax errors in the ASN.1 sections have been found:

Annex B.4

Add semicolon to end of EXPORTS line.

Corrected:

```
EXPORTS        NOTIFICATION, bqsigIeNotification;
```

Original:

```
EXPORTS        NOTIFICATION, bqsigIeNotification
```

Annex B.5

Add "SIZE" to OCTET STRING

Corrected:

```
NsapEncodedNumber      ::=      OCTET STRING (SIZE(20))
```

Original:

```
NsapEncodedNumber      ::=      OCTET STRING (20)
```

Annex B.8

Add semicolon to end of EXPORTS line.

Corrected:

```
EXPORTS                Extension, EXTENSION;
```

Original:

```
EXPORTS                Extension, EXTENSION
```

Annex C.1

In the definition of type ReturnResult, element result, change "@opcode" to "@.opcode" in the constraint.

Corrected:

```
        result    OPERATION.&ResultType
                      ({Operations}{@.opcode}
                      ! RejectProblem : returnResult-mistypedResult)
```

Original:

```
        result    OPERATION.&ResultType
                      ({Operations}{@opcode}
                      ! RejectProblem : returnResult-mistypedResult)
```

Annex C.4

Add semicolon to end of EXPORTS line.

Corrected:

```
EXPORTS        NOTIFICATION, bqsigIeNotification;
```

Original:

```
EXPORTS        NOTIFICATION, bqsigIeNotification
```

Annex C.4

Add "local:" before "2501".

Corrected:

```
bqsigIeNotification     NOTIFICATION::=
{                       ARGUMENT    BqsigInformationElement
                        CODE        local: 2501
```

Original:

```
bqsigIeNotification     NOTIFICATION::=
{                       ARGUMENT    BqsigInformationElement
                        CODE        2501
```

Annex C.5

Add "SIZE" to OCTET STRING

Corrected:

```
NsapEncodedNumber    ::=    OCTET STRING (SIZE(20))
```

Original:

```
NsapEncodedNumber    ::=    OCTET STRING (20)
```

Annex C.7

In the EXPORTS add "{}" after "NotificationDataStructure".

Corrected:

```
            EXPORTS          NotificationDataStructure{};
```

Original:

```
            EXPORTS          NotificationDataStructure;
```

Annex C.8

Add semicolon to end of EXPORTS line.

Corrected:

```
EXPORTS          Extension, EXTENSION;
```

Original:

```
EXPORTS          Extension, EXTENSION
```

Annex C.8

In the EXPORTS add "{}" after "Extension, EXTENSION".

Corrected:

```
            EXPORTS          Extension, EXTENSION{};
```

Original:

```
            EXPORTS          Extension, EXTENSION
```

Table of contents

1 Scope 1

2 Conformance 1

3 Normative references 1

4 Definitions 2

5 Abbreviations 2

6 Description 2
6.1 Overview 2
6.2 Addressing mechanisms 3
6.2.1 Explicit addressing 3
6.2.2 Functional addressing 3
6.3 Protocol architecture 3
6.4 Services provided by individual protocol entites 4
6.4.1 Services provided by ROSE 4
6.4.2 Services provided by GFT-control 4
6.4.3 Services provided by bearer-related transport 4
6.4.4 Services provided by connectionless bearer-independent transport 4
6.4.5 Services provided by connection-oriented bearer-independent transport 4

7 Operational requirements 4

8 Primitive definitions and state definitions 4
8.1 Primitive definitions 4
8.2 State definitions 5
8.2.1 APDU transport mechanisms 5
8.2.2 GFT-Control 5

9 Coding requirements 5
9.1 Message functional definitions and content 5
9.1.1 Additional messages for bearer-related transactions 5
9.1.2 Messages for connectionless bearer-independent transport 6
9.1.3 Messages for connection-oriented bearer-independent transport 6
9.2 General message format and information element coding 8
9.2.1 Message type 8
9.2.2 Other information elements 8
9.2.3 Encoding of information described using ASN.1 13

10 Signalling procedures 13
10.1 APDU transport mechanisms 13
10.1.1 Bearer-related transport 13
10.1.2 Bearer-independent transport mechanisms 14
10.1.3 Connection-oriented bearer-independent transport mechanism 14
10.1.4 Connectionless bearer-independent transport mechanism 15
10.2 GFT-Control procedures for APDUs 16

10.2.1 GFT-control procedures for transport of APDUs 16
10.2.2 GFT-Control procedures for CO-BI connection control 19
10.3 Remote operations procedures 20
10.3.1 Introduction 20
10.3.2 Procedures for operations 21
10.4 Notification transport mechanisms 21
10.4.1 Sending notification information 21
10.4.2 Receiving notification information 21
10.5 GFT-Control procedures for notifications 21
10.5.1 Actions at a PINX which generates notifications 21
10.5.2 Actions at a receiving PINX 21

11 Interworking with (narrowband) QSIG 22
11.1 Full termination of generic functional protocol 22
11.2 Generic interworking function 23
11.2.1 Architecture 23
11.2.2 Bearer-related transport mechanism 23
11.2.3 Connection-oriented bearer independent mechanism 24

12 Parameter values 25
12.1 Connection-oriented bearer-independent transport 25

13 Dynamic description (SDLs) 25
13.1 Block overview diagram 25
13.2 Component transport mechanisms 25
13.2.1 Bearer-related transport mechanism 25
13.2.2 Connection-oriented bearer-independent transport mechanism 25
13.2.3 Connectionlessbearer-independent transport mechanism 26
13.3 GFT-Control 26

14. Manufacturer Specific Information (MSI) 32
14.1 Manufacturer specific operations and errors 32
14.2 Manufacturer specific additions to standardised operations and error 32
14.3 Manufacturer specific notification 32

Annex A - Protocol implementation conformance statement (PICS) proforma 35

Annex B - Formal definition of data types using ITU-T Rec. X.208 63

Annex C - Formal definition of data types using ITU-T Rec. X.680 71

Annex D - Information flows 81

Annex E - Instruction indicators 83

Annex F - Formal definitions of remote operations notation using ITU-T Rec. X.208 85

Annex G - Formal definitions of remote operations notation using ITU-T Rec. X.680 87

Annex H - Examples of the use of Manufacturer Specific Information 91

Annex I - Remote operations protocol 95

Annex J - Problem code definitions 97

1 Scope

This Standard specifies the functional protocol for the support of supplementary services and additional basic call capabilities at the Q-reference point. The Q-reference point exists between Private Integrated Services Exchanges (PINX) connected together within a Private Integrated Services Network (PISN) and is defined in ISO/IEC 11579-1. The generic functional protocol is part of the B-QSIG signalling system.

The procedures specified in this Standard can be used in association with a bearer connection (bearer-related) or outside the context of any bearer connection (bearer-independent).The application of this Standard to individual additional basic call capabilities and supplementary services is outside the scope of this Standard and should be defined in those standards or proprietary specifications that specify the individual capabilities.

All conformance to this Standard is based on the external behaviour at the interface at Q-reference point, i.e. on the generation of the correct message structure and in the proper sequence as specified in this Standard.

The generic functional protocol is based on the DSS2 generic functional protocol specified ITU-T Rec. Q.2932.1 but extended to allow non local information exchange as well as local information exchange.

This Standard is applicable to PINXs supporting additional basic call capabilities and/or supplementary services requiring the functional protocol for signalling at the Q-reference point.

2 Conformance

In order to conform to this Standard, a PINX shall satisfy the requirement identified in the Protocol Implementation Conformance Statement (PICS) proforma in annex A.

3 References (normative)

The following standards contain provisions which, through reference in this text, constitute provision of this Standard. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

ISO/IEC 11579-1	Information technology - Telecommunications and information exchange between systems - Private Integrated Services Network - Part 1: Reference configuration for PISN exchanges (PINX) (1994)
ISO/IEC DIS 13246	Information technology - Telecommunications and information exchange between systems - Broadband Private Integrated Services Network -Inter-exchange signalling protocol - Signalling ATM adaptation layer (1996)
ISO/IEC DIS 13247	Information technology - Telecommunications and information exchange between systems - Broadband Private Integrated Services Network - Inter-exchange signalling protocol - Basic call/connection control (1996)
ITU-T Rec. E.164	Numbering plan for the ISDN era (1991)
ITU-T Rec. Q.2932.1	B-ISDN Digital Subscriber Signalling System No. 2 (DSS2); Generic Functional Protocol Part 1 (1996)
ITU-T Rec. X.208	Specification of Abstract syntax Notation One (ASN.1) (1993)
ITU-T Rec. X.209	Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1) (1993)
ITU-T Rec. X.213	Information technology - Network service definition for open systems interconnection (1992)
ITU-T Rec. X.229	Remote Operations: Protocol Specification (1993)
ITU-T Rec. X.680	Information Technology - Abstract Syntax Notation One (ASN.1): Specification of Basic Notation (1994)
ITU-T Rec. X.690	Information Technology - ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rues (CER) and Distinguished Encoding Rules (DER) (1994)
ITU-T Rec. X.880	Information Technology - Remote operations, Concepts, model and notation (1995)

4 Definitions

Definitions in ITU-T Rec. Q.2932.1 clause 3 shall apply with the following additions.

Adjacent PINX: A PINX as considered from another PINX to which it is directly connected via one or more IPLs.

Destination PINX: In the context of a single one-way exchange of information between two AS-Control entities the PINX where the receiving AS-Control entity is located.

End PINX: In the context of a particular call/connection or a particular CO-BI connection, an Originating or Terminating PINX. It can also be a Gateway PINX, dependent on the capabilities of the signalling system being interworked (i.e. unless it transports APDUs unchanged to or from the other signalling system.

Gateway PINX: The definition in ISO/IEC 13247 shall apply. Dependent on the capabilities of the signalling system being interworked by the gateway PINX, it can act as a Transit or an End PINX in the context of the supplementary services APDUs. That is, it can either transport the APDUs unchanged to or from the other signalling system, perhaps embedded in some other protocol unit, or process the APDUs and perform an interworking function of the information flows and encoding of the Supplementary service concerned.

Inter-PINX link (IPL): The definition in ISO/IEC 13247 shall apply.

Next PINX: An adjacent PINX to which an APDU is to be sent in the context of an existing signalling connection (related to a bearer or independent of a bearer).

Originating PINX: The definition in ISO/IEC 13247 shall apply. In addition, the term is also applied to a PINX which originates a CO-BI connection.

Preceding side: in the context of a call/connection or a CO-BI connection using an IPL, the side that initiates call/connection or CO-BI connection establishment over that IPL (see figure 1 in ISO/IEC 13247).

Private Integrated Services Network Exchange (PINX): as specified in ISO/IEC 11579-1.

Source PINX: In the context of a single one-way exchange of information between two AS-Control entities, the PINX where the sending AS-Control entity is located.

Succeeding side: in the context of a call/connection or a CO-BI connection using an IPL, the opposite side from the side that initiates call/connection or CO-BI connection establishment over that IPL (see figure 1 in ISO/IEC 13247).

Terminating PINX: The definition in ISO/IEC 13247 shall apply. In addition, the term is also applied to a PINX which terminates a CO-BI connection.

Transit PINX: The definition in ISO/IEC 13247 shall apply. In addition, the term is also applied to a PINX through which a CO-BI connection passes, excluding any Originating PINX, Terminating PINX or Gateway PINX.

5 Abbreviations

Abbreviations in ITU-T Rec. Q.2932.1 clause 4 shall apply with the following additions

IPL	Inter-PINX Link
MSI	Manufacturer Specific Information
PICS	Protocol Implementation Conformance Statement
PISN	Private Integrated Services Network
PINX	Private Integrated Network Exchange

6 Description

6.1 Overview

The generic functional protocol provides a means of exchanging ROSE APDUs on behalf of Application Service Control entities located in different PINXs. These Application Service Control entities may be for the support of supplementary services or additional basic call capabilities. This exchange may take place either in association with a bearer established using the procedures of ISO/IEC 13247 or independently of any bearer. Bearer independent transport can either be connection-oriented or connectionless. In the case of connection-oriented bearer-independent transport, establishment and release of the connection is specified in this Standard.

For bearer-related transport and connection-oriented bearer-independent transport, the exchange of ROSE APDUs can be between any of two PINXs involved in the connection, as determined by addressing information transported with the APDUs (e.g., between the two End PINXs). For connectionless bearer independent transport, the exchange of ROSE APDUs is between the source PINX and the destination PINX for the transporting message.

6.2 Addressing mechanisms

Communication between adjacent PINXs does not require addressing. Where the PINXs are not adjacent, addressing information is required in order to identify the receiving and sending AS-Control entities. Addressing may be in two forms:

- explicit addressing, where a number according to the numbering plan supported by the network is used to identify the receiving exchange and the sending exchange;
- functional addressing, where the recipient is identified by the function it is capable of supporting.

The addressing mechanisms are defined in a consistent manner for all transport mechanisms, but the capabilities that exist can be constrained by the particular transport mechanism used.

6.2.1 Explicit addressing

In explicit addressing the recipient is identified by a number which is assigned to the recipient.

Where the recipient is a PINX, then this can be a number assigned specifically for that purpose, or may be a number of some other addressable entity associated with that PINX.

The assigned number is according to the numbering plan of the PISN.

6.2.2 Functional addressing

The following functions are provided:

- End PINX (i.e. an originating, terminating or gateway PINX)
- Any Type of PINX (i.e. the next PINX that understands the contents).

6.3 Protocol architecture

Protocol Architecture described in ITU-T Rec. Q.2932.1 clause 5.2 shall apply with following modification:

- Replace all references to ITU-T Rec. Q.2931 and Q.2971 with references to ISO/IEC 13247.

Generic Functional Transport (GFT-) Control provides a means of transporting APDUs between two PINXs using one of the underlying transport mechanisms. In the case of the bearer-related and connection-oriented bearer-independent transport mechanisms, the two PINXs lie on the path of the call/connection or CO-BI connection and are not necessarily adjacent.

Figure 1 shows the application of the protocol model to the case where the AS-Control entities to be associated in communication are not in adjacent PINXs. In the example shown, communication is via a single intervening PINX. It may be generally applied to communication via more than one intervening PINX by simple replication. Other transport mechanisms could be used as described in 6.1.

In figure 1, relaying functions at the intervening node are performed by GFT-Control

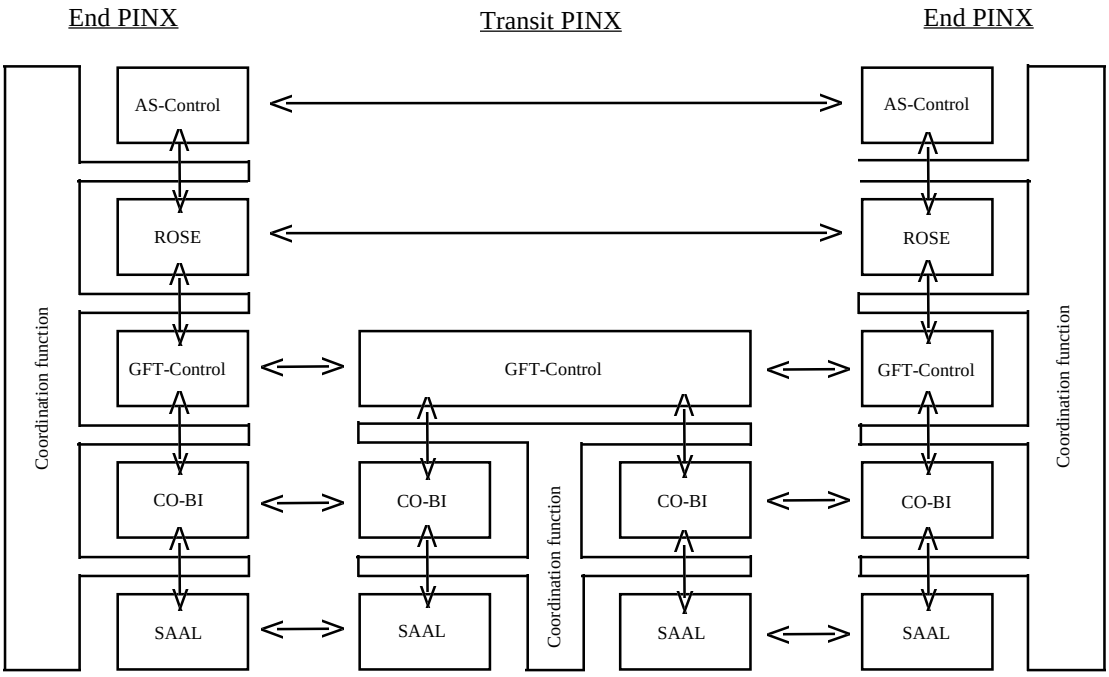


Figure 1 - Application of the protocol model to communication between non-adjacent PINXs

6.4 Services provided by individual protocol entites

6.4.1 Services provided by ROSE

Services provided by ROSE described in ITU-T Rec. Q.2932.1 clause 5.4.1 shall apply.

6.4.2 Services provided by GFT-control

Services provided by GFT-control described in ITU-T Rec. Q.2932.1 clause 5.4.2 shall apply.

6.4.3 Services provided by bearer-related transport

Services provided by bearer-related transport described in ITU-T Rec. Q.2932.1 clause 5.4.3 shall apply with following modification:

– Replace all references to ITU-T Rec. Q.2931 and Q.2971 with references to ISO/IEC 13247.

6.4.4 Services provided by connectionless bearer-independent transport

Services provided by connectionless bearer-independent transport described in ITU-T Rec. Q.2932.1 clause 5.4.4 shall apply.

6.4.5 Services provided by connection-oriented bearer-independent transport

Services provided by connection-oriented bearer-independent transport described in ITU-T Rec. Q.2932.1 clause 5.4.5 shall apply.

7 Operational requirements

The requirements for provision of this Standard are dependent on the needs of the applications that are to exchange information. In particular, support of each individual transport mechanism is optional, although at least one transport mechanism shall be supported.

8 Primitive definitions and state definitions

8.1 Primitive definitions

Primitive definitions described in ITU-T Rec. Q.2932.1 clause 7.1 shall apply with following modification:

– Replace all references to ITU-T Rec. Q.2931 and Q.2971 with references to ISO/IEC 13247.

8.2 State definitions

8.2.1 APDU transport mechanisms

8.2.1.1 Bearer-related transport mechanism

There are no additional call/connection states over and above those defined in ISO/IEC 13247 clause 6.4 .

8.2.1.2 Connectionless bearer-independent transport mechanism

Connectionless bearer-independent transport states described in ITU-T Rec. Q.2932.1 clause 7.2.1.2 shall apply.

8.2.1.3 Connection-oriented bearer-independent transport mechanism

Connection-oriented bearer-independent transport states described in ITU-T Rec. Q.2932.1 clause 7.2.1.3 shall apply.

8.2.2 GFT-Control

The GFT-control state described in ITU-T Rec. Q.2932.1 clause 7.2.2 shall apply.

In addition, the following states shall apply for the control of CO-BI connections.

- Originating PINX GFT-Control States:
  - Originating_connection_idle: no connection exists
  - Originating_connection_request: connection establishment has been requested, but no response has been received from the Terminating PINX.
  - Originating_connection_active: the connection is active
- Transit PINX GFT-Control States:
  - Transit_connection_idle: no connection exists
  - Transit_connection_request: connection estab-lishment request has been received from the Pre-ceding PINX and forwarded to the Succeeding PINX, but no response has been received from the Succeeding PINX.
  - Transit_connection_active: the connection is active
- Terminating PINX GFT-Control States:
  - Incoming_connection_idle: no connection exists
  - Incoming_connection_active: the connection is active

9 Coding requirements

9.1 Message functional definitions and content

This subclause shall be read in conjunction with clause 7 of ISO/IEC 13247. All messages are additional to those defined in that clause and the following tables should be interpreted according to the introductory material of clause 7 of ISO/IEC 13247.

In the following subclauses the key as described below applies to the "Reference" column.

Key: ISO/IEC 13247 /nn: reference to subclause nn in ISO/IEC 13247,
B-QSIG-GF /nn: reference to subclause nn in this Standard.

To determine if an information element specified in this Standard is allowed to be included in the following messages, see subclause 9.2.

Information elements not defined in subclause 9.2 are only allowed to be included in the following messages when explicitly indicated in the message structure.

9.1.1 Additional messages for bearer-related transactions

In addition to the message structures defined below, the Facility information element may also be included in any of the following messages: ALERTING, CONNECT, PROGRESS, RELEASE, RELEASE COMPLETE,

SETUP, ADD PARTY, ADD PARTY ACKNOWLEDGE, PARTY ALERTING, ADD PARTY REJECT, DROP PARTY AND DROP PARTY ACKNOWLEDGE.

9.1.1.1 FACILITY

This message may be sent by the preceding side or the succeeding side to control a supplementary service or additional basic call capability. The supplementary service or additional basic call capability to be invoked, and its associated parameters, are specified in the Facility information element.

The structure of the FACILITY message is shown in table 1/Q.2932.1 with the modification that the maximum length and the maximum number of repetition of the Notification indicator information element are implementation options and with the modification that the Endpoint reference information element shall be included in the case of point-to-multipoint connections.

9.1.2 Messages for connectionless bearer-independent transport

9.1.2.1 FACILITY

This message may be sent between two adjacent PINXs to control a supplementary service or additional basic call capability. The supplementary service or additional basic call capability to be invoked, and its associated parameters, are specified in the Facility information element.

The structure of the FACILITY message is shown in table 2/Q.2932.1 with the following modification:

– Called party number is not applicable.
– Called party subaddress is not applicable.
– Calling party number is not applicable.

The Facility Information element may be repeated any number of times

The maximum length and the maximum number of repetition of the Notification indicator information element are implementation options.

9.1.3 Messages for connection-oriented bearer-independent transport

9.1.3.1 CALL PROCEEDING

This message is sent by the preceding side to the succeeding side or by the succeeding side to the preceding side, to indicate that the requested transport establishment has been initiated and that no more establishment information will be accepted.

The structure of the CALL PROCEEDING message is the same as that shown in ISO/IEC 13247 subclause 7.1.2.

9.1.3.2 CO-BI SETUP

This message is sent by the preceding side to the succeeding side, to initiate transport establishment. The structure of the CO-BI SETUP message is shown in table 1:

Table 1 - CO-BI SETUP message content

Message type: CO-BI SETUP
Direction: Preceding to succeeding

Information element	Reference	Type	Length
Protocol discriminator	ISO/IEC 13247/8.2	M	1
Call reference	ISO/IEC 13247/8.3	M	4
Message type	ISO/IEC 13247/8.4	M	2
Message length	ISO/IEC 13247/8.4	M	2
Facility	B-QSIG-GF /9.2.2.2	O (NOTE 1)	4-*
Called party number	ISO/IEC 13247/8.5	M	4-*
Calling party number	ISO/IEC 13247/8.5	M	4-*
Notification indicator	B-QSIG-GF /9.2.2.3	O (NOTE 2)	4-*

NOTE 1 – Included if the requesting GFT-Control wishes to include APDUs in the setup request. This information element may be repeated any number of times.

NOTE 2 – This indicator may be present whenever notification is delivered. The Notification indicator information element may be repeated in this message. The maximum length and the maximum number of repetitions allowed are implementation options.

9.1.3.3 CONNECT

This message is sent by the succeeding side to the preceding side to indicate acceptance of a transport establishment request by the called entity.

The structure of the CONNECT message is shown in table 5/Q.2932.1 with the modification that the maximum length and the maximum number of repetition of the Notification indicator information element are implementation options.

9.1.3.4 FACILITY

This message may be sent by the preceding or the succeeding side to control a supplementary service or additional basic call capability. The supplementary service or additional basic call capability to be invoked, and its associated parameters, are specified in the Facility information element.

The structure of the FACILITY message is shown in table 1/Q.2932.1 with the following modification:

– Called party number is not applicable.
– Called party subaddress is not applicable.
– Calling party number is not applicable.

The Facility Information element may be repeated any number of times. The maximum length and the maximum number of repetitions of the Notification indicator information element are implementation options.

9.1.3.5 NOTIFY

This message is sent by the preceding side or the succeeding side to indicate information pertaining to a call/connection. The structure of the NOTIFY message is the same as that shown in ISO/IEC 13247 subclause 7.1.7.

9.1.3.6 RELEASE

This message is sent by the transport entity to request clearing of the part of the end-to-end transport connection controlled by the peer transport entity and to prepare to release its call reference value after sending RELEASE COMPLETE.

The structure of the RELEASE message is shown in table 6/Q.2932.1 with the modification that the maximum length and the maximum number of repetitions of the Notification indicator information element are implementation options.

9.1.3.7 RELEASE COMPLETE

This message is sent by the preceding side or the succeeding side to indicate that the transport entity sending the message has released its call reference value. The receiving equipment shall release its call reference value.

The structure of the RELEASE COMPLETE message is the same as that shown in ISO/IEC 13247 subclause 7.1.5 (see table 6/ISO/IEC 13247).

9.1.3.8 STATUS

This message is sent from the preceding side or the succeeding side in response to a STATUS ENQUIRY message or at any point in time to report certain error conditions. The structure of the STATUS message is the same as that shown in ISO/IEC 13247 subclause 7.1.

9.1.3.9 STATUS ENQUIRY

This message is sent by the preceding side or the succeeding side at any time to solicit a STATUS message from the peer layer 3 entity. Sending a STATUS message in response to a STATUS ENQUIRY message is mandatory. The structure of the STATUS ENQUIRY message is as shown in ISO/IEC 13247 subclause 7.1.

9.2 General message format and information element coding

Clause 8 of ISO/IEC 13247 shall apply with the following additions.

9.2.1 Message type

The additional message type codings for the purpose of this Standard are defined in table 7/Q.2932.1.

9.2.2 Other information elements

Table 8/Q.2932.1 shows the additional information elements defined for the generic functional protocol.

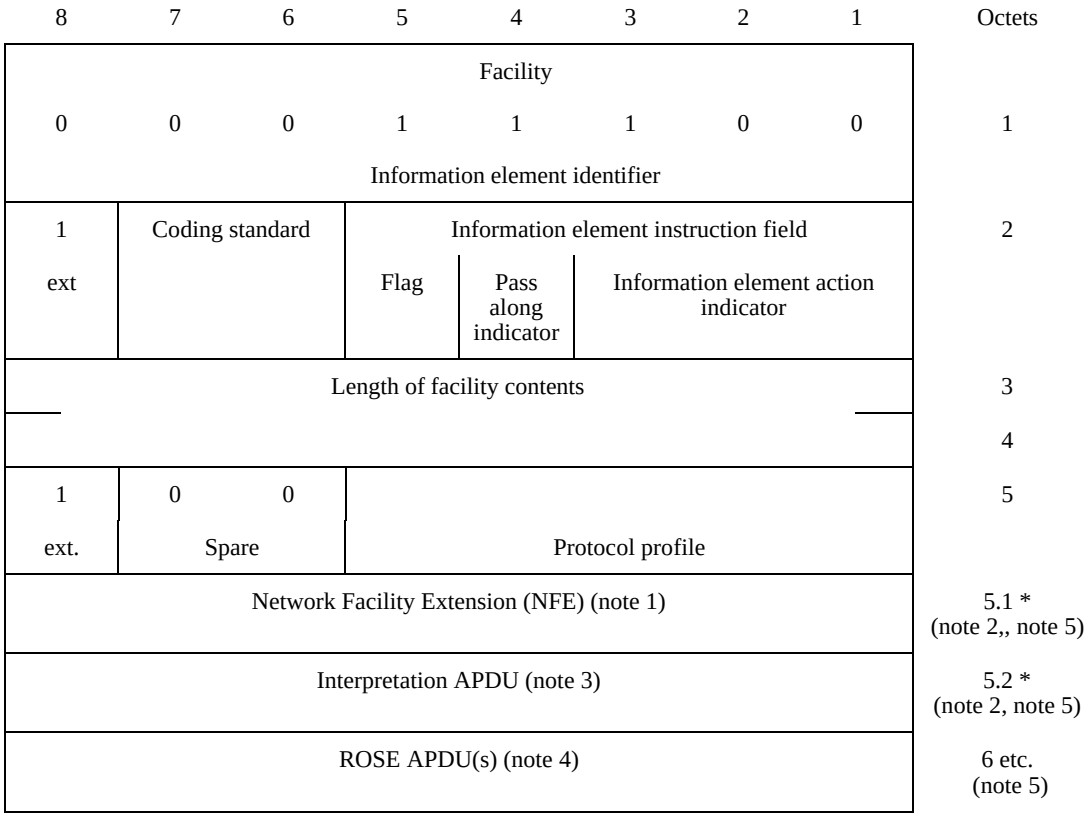
9.2.2.1 Call state

The call state information element is defined as in subclause 8.5.15 of ISO/IEC 13247. However the state value assignments defined in table 9/Q.2932.1 exist for the connection-oriented bearer-independent transport mechanism.

9.2.2.2 Facility

The purpose of the Facility information element is to convey an optional interpretation APDU and one or more ROSE APDUs.

Figure 2 shows the structure of the Facility information element. Table 2 shows the value of the protocol profile field applicable for supplementary services or additional basic call capabilities.



Note 1. The Network Facility Extension (NFE), as defined in clause 9.2.2.2.1, may be included, in accordance with the procedures of clause 10.2

Note 2. Each of octets groups 5.1 and 5.2 comprises an ASN.1 type encoded as defined clause 9.2.3 The presence or absence of each of these octets groups can be determined from the presence or absence of the tag values concerned in the appropriate position in the Facility information element.

Note 3. The interpretation APDU, as defined in clause 9.2.2.2.2, may be included, in accordance with the procedures in 10.2.

Note 4. One or more ROSE APDUs in accordance with 9.2.2.2.3 may be included depending on specific service requirements. Multiple ROSE APDUs may be sent in one Facility information element or in more than one (individual) Facility information elements, separate Facility information elements shall be used if different values apply for the NFE or the Interpretation APDU. Otherwise it is a sender's choice to use either one or several Facility information elements taking into account the maximum length of the Facility information element.

Note 5. The NFE (if present), the Interpretation APDU (if present) and the ROSE APDU(s) are collectively known as the generic functional data.

Figure 2 - Facility information element

Table 2 - Facility information element protocol profile

Bits 5 4 3 2 1	
1 1 1 1 1	Networking extensions (Note 1)
All other values are reserved and their usage is the subject of other standards	
Note 1 - ISO/IEC defined local values apply in the ROSE APDUs. These local values are specified in ISO/IEC standards using this Standard.	

9.2.2.2.1 Network Facility Extension (NFE)

The NFE shall comprise ASN.1 type NetworkFacilityExtension as defined in B.3 of annex B using ASN.1 as specified in ITU-T recommendation X.208 and in C.3 of annex C using ASN.1 as specified in ITU-T recommendation X.680, encoded in accordance with clause 9.2.3. This provides a means of routing the contents of the Facility information element within the context of a call/connection or a CO-BI connection across the PISN, and a means of identifying the origin and destination of the information, in accordance with the procedure of clause 10.2.

Clause 10.2 describes the use of the particular elements of NFE.

9.2.2.2.2 Interpretation APDU

The interpretationAPDU shall comprise ASN.1 type InterpretationAPDU as defined in B.6 of annex B using ASN.1 as specified in ITU-T recommendation X.208 and in C.6 of annex C using ASN.1 as specified in ITU-T recommendation X.680, encoded in accordance with clause 9.2.3. This APDU provides a means whereby the originator can include optional instructions to the receiving PINX for use in the event that it does not understand the operation value of an invoke APDU contained in octet 6 onwards of the Facility information element.

Clause 10.2 will describe the use of the interpretationAPDU.

9.2.2.2.3 ROSE APDU

A ROSE APDU shall comprise ASN.1 type APDU as defined in B.1 of annex B using ASN.1 as specified in ITU-T recommendation X.208 and in C.1 of annex C using ASN.1 as specified in ITU-T recommendation X.680, encoded in accordance with 9.2.3.

In accordance with X.229 and X.880, ROSE APDUs are of four types:

- Invoke APDU (ASN.1 type InvokeAPDU, based on ROIV-APDU in X.229, or ASN.1 type Invoke based on Invoke in X.880).
- Return result APDU (ASN.1 type ReturnResultAPDU, based on RORR-APDU in X.229, or ASN.1 type ReturnResult, based on ReturnResult in X.880).
- Return error APDU (ASN.1 type Return ErrorAPDU, based on RORE-APDU in X.229, or ASN.1 type ReturnError, based on ReturnError in X.880).
- Reject APDU (ASN.1 type RejectAPDU, based on RORJ-APDU in X.229, or ASN.1 type Reject, based on Reject in X.880).

Note 1. The definition of types InvokeAPDU, ReturnResultAPDU, ReturnErrorAPDU and RejectAPDU in table B-1 and types Invoke, ReturnResult, ReturnError and Reject in table C-1 are equivalent to the corresponding definitions in clause 9 of X.229 and X.880 respectively, with the exception that a number of the ASN.1 types in table B-1 and C-1 (e.g. InvokeIdType) are size delimited to enhance interoperability in a multivendor B-PISN.

Note 2. Annex I gives a general overview of the ROSE protocol and its constituent parts. Annex J provides definitions of the problem codes for use in type RejectAPDU.

Invoke APDUs, return result APDUs and return error APDUs used in the context of a supplementary service or additional basic call capability will be implicitly defined by the operations and errors used by that supplementary service or additional basic call capability. These operations and errors will be defined using ASN.1 in the relevant supplementary service or additional basic call capability specifications (standardised or manufacturer specific).

9.2.2.3 Notification indicator

The purpose of the notification indicator information element is to convey a notification.

The Notification indicator information element is coded as shown in figure 3 and table 3, this being an extension of the coding specified in ISO/IEC 13247.

The maximum length of the information element is application dependent.

8	7	6	5	4	3	2	1	Octets
Notification Indicator								
0	0	1	0	0	1	1	1	1
Information element identifier								
1 ext	Coding standard		Information element instruction field					2
			Flag	Pass along indicator.	Information element action indicator			
Length of notification element contents								3
								4
1	Notification Description Encoding							5
ASN.1 encoded Notification Data Structure								5.1 (note)

Note. Octet 5.1 shall only be included when the notification description indicates "discriminator for notification extension" or "discriminator for extension to ISO defined ASN.1 encoded notification data structure".

Figure 3 - Notification indicator information element

Table 3 - Notification Description encoding

7 6 5 4 3 2 1	
0 0 0 0 0 0 0	reserved for notification values assigned by ITU-T
through	
0 0 0 0 0 1 0	
0 0 0 0 0 1 1	discriminator for notification extension
0 0 0 0 1 0 0	reserved for notification values assigned by ITU-T
through	
0 0 1 1 1 1 1	
0 1 0 0 0 0 0	reserved for notifications values assigned by ISO
through	
0 1 1 1 1 1 1	
1 0 0 0 0 0 0	discriminator for extension to ISO defined ASN.1 encoded notification data structure
1 0 0 0 0 0 1	reserved for notification values assigned by ITU-T
through	
1 1 1 1 1 1 1	
all values shall be treated as valid	

A notification can be either a simple notification comprising only an integer value in octet 5 or an ASN.1-encoded notification in octet(s) 5.1. In the latter case octet 5 contains either the value “discriminator for notification extension” or “discriminator for extension to ISO defined ASN.1 encoded notification data structure”. An ASN.1-encoded notification is defined using the NOTIFICATION macro specified in B.4 of annex B using ASN.1 as specified in ITU-T recommendation X.208 or using the NOTIFICATION object class specified in C.4 of annex C using ASN.1 as specified in ITU-T recommendation X.680.

Notification Description value “discriminator for notification extension” shall be used for notifications defined using ASN.1 in which the notification value is either of type INTEGER with a value defined by ITU-T or of type OBJECT IDENTIFIER. Notification values of type OBJECT IDENTIFIER include manufacturer specific notifications (see 14.3). Notification Description value “discriminator for extension to ISO defined ASN.1 encoded notification data structure” shall be used for notifications defined using ASN.1 in which the notification value is of type INTEGER with a value defined by ISO. In either case, octet 5.1 shall contain ASN.1 type NotificationDataStructure, as defined in B.7 of annex B using ASN.1 as specified in ITU-T recommendation X.208 and in C.7 of annex C using ASN.1 as specfied in ITU-T recommendation X.680, encoded in accordance with 9.2.3. Element notificationTypeID shall contain the notification value and element notificationArgument shall contain any additional data.

B.4 of annex B and C.4 of annex C also define the ASN.1-encoded notification bqsigIeNotification, which can be used to convey B-QSIG information elements as a notification. Other notifications will be defined using the NOTIFICATION macro in the relevant supplementary specifications (standardised or manufacturer specific).

9.2.2.4 Treatment of existing ISO/IEC 13247 information elements as parameters

Supplementary service or additional basic call capability protocol specifications are expected to require new parameters to be defined and to require existing ISO/IEC 13247 information elements.

New parameters shall be defined using ITU-T Rec. X.209 coding, or ITU-T Rec. X.690 as appropriate, if they do not appear elsewhere in ISO/IEC 13247 messages.

Supplementary service or additional basic call capability protocol specifies may elect to encapsulate one or more existing ISO/IEC 13247 information elements within an ITU-T Rec. X.209 data element, or ITU-T Rec. X.690 data element, as appropriate, thereby retaining the ISO/IEC 13247 coding for these information elements. When this option is chosen, all the ISO/IEC 13247 information elements should be grouped together as the content following the BqsigInformationElement tag. This data element may appear by itself or as a member of a sequence or set.

Encapsulation of the Facility information element within Facility information elements shall not be used.

Type BqsigInformationElement is defined in B.2 of annex B using ASN.1 as specified in ITU-T recommendation X.208 and in C.2 of annex C using ASN.1 as specified in ITU-T recommendation X.680.

9.2.3 Encoding of information described using ASN.1

When specified according to ITU-T Rec. X.208, all data structures in the Facility information element (octet 5.1 onwards) and in the Notification indicator information element (octet 5.1) shall be encoded according to the Basic Encoding Rules (BER) as specified in ITU-T Rec. X.209.

When specified according to ITU-T Rec. X.680, all data structures in the Facility information element (octet 5.1 onwards) and in the Notification indicator information element (octet 5.1) shall be encoded according to the BER as specified in ITU-T Rec. X.690.

The following guidelines apply for the application of the different length encodings:

- the short form definitive length encoding should be used to indicate the length of a data value with a length less than 128 octets;
- when the long form definitive length encoding is used, the minimum number of octets should be used;
- OCTET STRING and BIT STRING values should be encoded in a primitive form.

Receiving entities shall be able to interpret all length forms of the basic encoding rules.

10 Signalling procedures

10.1 APDU transport mechanisms

The transport function for operations is performed by the exchange of APDUs via B-QSIG messages.

A supplementary service or additional basic call capability functional protocol (using the Facility information element) may use an existing bearer-related call reference if it is to be coupled to the connection, or it may use a bearer-independent call reference.

10.1.1 Bearer-related transport

The definition of "Bearer-related transport mechanism" is given in ITU-T Rec. Q.2932.1 clause 3.

The procedures for call/connection control are described in ISO/IEC 13247, clause 9 and 10. These procedures are not influenced by the APDUs carried. Bearer-related transport procedures and operations shall operate independently of each other.

10.1.1.1 Normal operation

For bearer-related transport any message in which the Facility information element may be included (see 9.1.1) may be used to carry the APDUs. These messages shall use the call reference of the bearer connection.

The FACILITY message shall not be sent in the following call/connection states:

- Null (0)
- Call Initiated (1)
- Call Present (6)
- Release Request (11)
- Release Indication (12)

The call reference provides the means to correlate messages belonging to the same connection. When a supplementary service or additional basic call capability affects more than one connection, different call

references are used to identify each connection individually. This implies the use of different messages in order to manage each connection separately.

When the call/connection associated with the AS-Control functionality is cleared due to AS-Control actions, the Cause information element in the clearing message shall be set to #16 "normal clearing".

Any additional reason for clearing is included in the APDUs generated by AS-Control, and therefore transferred in the Facility information element.

When indicated by GFT-Control, generic functional data and a protocol profile value shall be included in a Facility information element and transferred in a call control message or party control message if such a message is being sent for other reasons, or else in a FACILITY message.

The transport mechanism shall pass all valid received generic functional data and protocol profile values in the Facility information element to GFT-Control and the procedures specified in GFT-Control (see subclause 10.2) shall also apply.

10.1.1.2 Exceptional procedures

If a receiving entity recognises a supplementary service or additional basic call capability request in a received SETUP message but is not able to process the request, then the following options shall apply:

– the receiving entity may clear the call/connection request and reject the supplementary service or additional basic call capability invocation by means of an appropriate call-clearing message which contains the Cause information element and a return error APDU with the appropriate parameters in the Facility information element;

– the receiving entity may continue to process the call/connection request according to the call/connection control procedures of ISO/IEC 13247, and reject the supplementary service or additional basic call capability invocation by including a return error APDU with the appropriate parameters in the Facility information element by means of a FACILITY message or in an appropriate call/connection control message or party control message;

The option to be used depends on the individual supplementary service or additional basic call capability procedures which are the subject of other standards.

In addition, when the receiving entity identifies an error in the received APDU, the receiving entity may continue to process the call/connection request according to the call/connection control procedures of ISO/IEC 13247, and ignore the supplementary service or additional basic call capability invocation, in which case a reject component shall be generated.

No response message shall be sent after the call reference value has been released.

The procedures of subclause 10.1.1 are an extension to the procedures of ISO/IEC 13247. As such the general error handling procedures as defined in subclause 9.6 of ISO/IEC 13247 apply. However, the handling of errors in octets 5 onwards of the Facility information element is specified in subclause 10.2.1. The handling of errors in APDUs is specified in subclause 10.3 If the connection is being cleared, the treatment of outstanding supplementary service or additional basic call capability requests is subject to the standards for the individual supplementary services or additional basic call capabilities.

10.1.2 Bearer-independent transport mechanisms

Bearer-independent transport mechanism described in ITU-T Rec. Q.2932.1 clause 9.1.2 shall apply with following modification:

– B-QSIG utilises the signalling AAL connection defined in ISO/IEC 13246 instead of ITU-T Rec. Q.2130.

10.1.3 Connection-oriented bearer-independent transport mechanism

Connection-oriented bearer-independent transport mechanism described ITU-T Rec. Q.2932.1 clause 9.1.3 shall apply.

10.1.3.1 Actions in the Null state

Actions in the null state described in ITU-T Rec. Q.2932.1 clause 9.1.3.1 shall apply with following modification:

– Replace all references to ITU-T Rec. Q.2931 with references to ISO/IEC 13247.

– B-QSIG utilises the signalling AAL connection defined in ISO/IEC 13246 instead of ITU-T Rec. Q.2130.

– When entering the call present state, a CALL PROCEEDING message shall be sent.

– The preceding side shall include the Called party number and the Calling party number information elements identifying the destination and the source respectively of the bearer independent signalling connection.

10.1.3.2 Actions in the Call Present state

Actions in the call present state described in ITU-T Rec. Q.2932.1 clause 9.1.3.2 shall apply with the following modification:

– The sending of the CALL PROCEEDING message on request of GFT-Control is not applicable.

10.1.3.3 Actions in the Call Initiated state

Actions in the call initiated state described in ITU-T Rec. Q.2932.1 clause 9.1.3.3 shall apply.

10.1.3.4 Actions in the Incoming Call Proceeding state

Actions in the incoming call proceeding state described in ITU-T Rec. Q.2932.1 clause 9.1.3.4 shall apply.

10.1.3.5 Actions in the Outgoing Call Proceeding state

Actions in the outgoing call proceeding state described in ITU-T Rec. Q.2932.1 clause 9.1.3.5 shall apply.

10.1.3.6 Actions in the Active state

Actions in the active state described in ITU-T Rec. Q.2932.1 clause 9.1.3.6 shall apply.

10.1.3.7 Connection release

Connection release described in ITU-T Rec. Q.2932.1 clause 9.1.3.7 shall apply.

10.1.3.8 Actions in the Release Request state

Actions in the release request state described in ITU-T Rec. Q.2932.1 clause 9.1.3.8 shall apply.

10.1.3.9 Transport of APDUs associated with a connection-oriented bearer-independent signalling connection

Transport of APDUs associated with connection-oriented bearer-independent signalling connection described in ITU-T Rec. Q.2932.1 clause 9.1.3.9 shall apply.

10.1.3.10 Protocol error handling

Protocol error handling described in ITU-T Rec. Q.2932.1 clause 9.1.3.11 shall apply with following modification:

– Replace "- Actions regarding the handling of VCIs and VPCIs are not applicable;" with "- Actions regarding the handling of IPVCIs and IPVPIs are not applicable;".

– Replace all references to ITU-T Rec. Q.2931 with references to ISO/IEC 13247.

10.1.4 Connectionless bearer-independent transport mechanism

Connectionless bearer-independent transport mechanism described in ITU-T Rec. Q.2932.1 clause 9.1.4 shall apply with following modification:

– The connectionless bearer-independent transport mechanism is limited to local addressing only.

– The NOTE does not apply.

– Replace the reference to ITU-T Rec. Q.2931 with a reference to ISO/IEC 13247.

– B-QSIG utilises the signalling AAL connection defined in ISO/IEC 13246 instead of ITU-T Rec. Q.2130.

10.1.4.1 Normal operation

Normal operation described in ITU-T Rec. Q.2932.1 clause 9.1.4.1 shall apply.

10.1.4.2 Exceptional procedure

Exceptional procedure described in ITU-T Rec. Q.2932.1 clause 9.1.4.2 shall apply.

10.2 GFT-Control procedures for APDUs

10.2.1 GFT-control procedures for transport of APDUs

10.2.1.1 Actions at a source PINX

When ROSE or any other ASE requires to transmit generic functional data (i.e., one or more APDUs), this is indicated to GFT-Control. GFT-Control shall:

– determine from the information supplied by ROSE or any other ASE the transport mechanism required;

– ensure that the required transport mechanism is in a state to transmit generic functional data;

– supply to the protocol entity of the appropriate transport mechanism the generic functional data and protocol profile based on the type of ASE requesting transport of generic functional data (i.e., a protocol profile of "Networking extensions" denoting ROSE using local values specified by ISO/IEC);

– indicate the instruction indicator for use in the Facility information element.

NOTE
The prime function of the instruction indicator in the Facility information element is to provide corrective action when the generic functional protocol is not supported.

If GFT-Control is unable to provide the transfer of generic functional data, it shall indicate this to the ASE.

APDUs may be of two basic types:

– those which have only link significance, i.e. over a single IPL between two adjacent PINXs (local information exchange) ; or,

– those which have network significance, between two PINXs in the PISN which are not necessarily adjacent, and which can be, but need not be, the end PINXs involved in the call (non-local information exchange).

If the connectionless bearer-independent transport mechanism is used, only local information exchange shall be used.

For local information exchange, the Network Facility Extension (NFE), defined in 9.2.2.2.1, shall not be included in the generic functional data.

For non-local information exchange, the NFE shall be included, encoded as described in table 4.

NOTE
The generic functional data may contain one or more APDUs. If more than one APDU is contained in the generic functional data , they will be sent in a single Facility information element and will all be processed by the same Destination PINX. Any relationship between such APDUs is beyond the scope of this Standard.

Table 4 - Encoding of NFE

Case no.	Communication between	Required coding of NFE for each identified case			
		Encoding of sourceEntity	**Encoding of sourceEntityAddress**	**Encoding of destinationEntity**	**Encoding of destinationEntity-Address**
1	End PINX to End PINX	endPINX (Note)	Not included	endPINX	Not included
2	End PINX to addressed PINX	endPINX (Note)	Not included	anyTypeOfPINX	PINX address
3	End PINX to next PINX that understands contents	endPINX (Note)	Not included	anyTypeOfPINX	Not included
4	Transit PINX to End PINX	anyTypeOfPINX	PINX address	endPINX	Not included
5	Transit PINX to addressed PINX	anyTypeOfPINX	PINX address	anyTypeOfPINX	PINX address
6	Transit PINX to next PINX that understands contents	anyTypeOfPINX	PINX address	anyTypeOfPINX	Not included
NOTE. The value endPINX for the sourceEntity should be avoided if there is a possibility that the PINX can cease to be an End PINX (e.g., through the use of certain supplementary services) prior to a response (e.g., a Reject APDU) being received.					

If a Source PINX wishes to include additional information to facilitate handling of unrecognised ROSE APDUs of type InvokeAPDU at a Destination PINX, it shall include an Interpretation APDU (see 9.2.2.2.2) as the first APDU in the generic functional data sent to the protocol entity. If the NFE is included, the Interpretation APDU shall follow the NFE.

10.2.1.2 Actions at a receiving PINX

When GFT-Control receives generic functional data from the CL-BI transport mechanism, the PINX shall become the Destination PINX for that generic functional data.

When GFT-Control receives generic functional data from the bearer-related transport mechanism or the CO-BI transport mechanism it shall determine whether it is the Destination PINX for that generic functional data by checking for the presence of an NFE (by reference to the tag value of the first element in the generic functional data).

If the generic functional data does not contain an NFE, the PINX shall become the Destination PINX for that generic functional data.

If the generic functional data contains an NFE, the PINX shall determine whether it is a Transit PINX or End PINX in the context of the call/connection or CO-BI connection and act as described below.

10.2.1.2.1 End PINX actions

If the receiving PINX is an End PINX, and the encoding of the received NFE complies with the encoding and structure defined in 9.2.2.2.1, the following actions shall apply:

– if the destinationEntity element of the NFE indicates endPINX or anyTypeOfPINX and no destinationEntityAddress element is included, it shall become the Destination PINX for that generic functional data;

– if the destinationEntity element of the NFE indicates anyTypeOfPINX and the NFE includes a destinationEntityAddress element containing an address that matches the PINX's own address, the PINX shall become the Destination PINX for that generic functional data;

– if the destinationEntity element of the NFE indicates endPINX and erroneously includes a destinationEntityAddress element, the PINX shall become the Destination PINX for that generic functional data;

– if the destinationEntity element of the NFE indicates a value in the range 2 to 11, the receiving PINX shall become the Destination PINX for that generic functional data;

NOTE
Values 2 to 11 are reserved for future use. The behaviour specified above provides a measure of forward compatibility with anticipated uses of these reserved values, e.g., for addressing a terminal or a network edge.

– in all other cases, the received generic functional data shall be discarded.

If the received NFE does not conform to the encoding and structure defined in clause 9.2.2.2.1, the entire Facility information element shall be discarded.

10.2.1.2.2 Transit PINX actions

If the receiving PINX is a Transit PINX, and the encoding of the received NFE complies with the encoding and structure defined in 9.2.2.2.1, the following actions shall apply:

– if the destinationEntity element of the NFE indicates anyTypeOfPINX and the NFE includes a destinationEntityAddress element containing an address that matches the PINX's own address, the PINX shall become the Destination PINX for that generic functional data;

– if the destinationEntity element of the NFE indicates anyTypeOfPINX and no destinationEntityAddress element is included, the PINX may become the Destination PINX for that generic functional data if it understands the contents;

– if the destinationEntity element of the NFE indicates endPINX and erroneously includes a destinationEntityAddress element, the PINX shall ignore the contents of the destinationEntityAddress field and treat the contents of the generic functional data as if only the destinationEntity element was present;

– if the destinationEntity element of the NFE indicates endPINX, and the Transit PINX is capable of acting as an End PINX for all services indicated in the generic functional data, it may become the Destination PINX for that generic functional data;

NOTE
In this case, the source of the information will have no knowledge that the information has been intercepted, as the Transit PINX will act as if it were an End PINX. This may occur, for example, when a PINX at a PISN numbering domain boundary wishes to translate numbering information contained within an APDU.

– if the destinationEntity element of the NFE indicates a value in the range 2 to 11, and the Transit PINX is capable of acting as an End PINX for all services indicated in the generic functional data, it may become the Destination PINX for that generic functional data;

NOTE
Values 2 to 11 are reserved for future use. The behaviour specified above provides a measure of forward compatibility with anticipated uses of these reserved values, e.g., for addressing a terminal or a network edge.

– in all cases where the PINX does not become the Destination PINX, the generic functional data shall be passed on unchanged to the Next PINX.

If the received NFE does not conform to the encoding and structure defined in 9.2.2.2.1, the entire generic functional data shall be discarded and no generic functional data shall be passed on to the Next PINX.

10.2.1.3 Actions at a destination PINX

GFT-Control shall check the protocol profile, and if it is valid it shall indicate the generic functional data to the appropriate ASE, i.e., to ROSE if the protocol profile value is "Networking extensions". If the protocol profile value is a reserved value, the generic functional data shall be discarded and the procedures for unrecognized information element content specified in 9.6 of ISO/IEC 13247 shall be followed on the appropriate transport mechanism.

The generic functional data shall be discarded if octets beyond the NFE (if present) do not comprise one or more concatenated APDUs, each in the form of an encoded ASN.1 value (comprising tag, length and contents).

If the first APDU is an Interpretation APDU, GFT-Control shall examine any ROSE APDU of type RejectAPDU generated by ROSE as a result of the processing of these APDUs. If the element problem in the RejectAPDU is of type InvokeProblem and has value unrecognisedOperation the action taken shall depend on the contents of the Interpretation APDU as follows:

- if the Interpretation APDU indicates rejectUnrecognisedInvokePdu the ROSE APDU of type RejectAPDU shall be delivered to the destination indicated by ROSE;
- if the Interpretation APDU indicates clearCallIfAnyInvokePduNotRecognised the ROSE APDU of type RejectAPDU shall be delivered to the destination indicated by ROSE, and the transport mechanism shall be requested to clear the call/connection or the CO-BI connection to which the InvokeAPDU was related.
- if the Interpretation APDU indicates discardAnyUnrecognisedInvokePDU the ROSE APDU of type RejectAPDU shall be discarded.

If no Interpretation APDU is received, any ROSE APDUs of type RejectAPDU shall be delivered to the destination indicated by ROSE.

10.2.2 GFT-Control procedures for CO-BI connection control

10.2.2.1 Actions at an Originating PINX

10.2.2.1.1 Actions in the Originating_connection_idle state

When a request for establishment of a CO-BI connection to a remote PINX is received from an ASE, GFT-Control shall request the Outgoing side protocol entity to send a CO-BI SETUP message, including the address of the Terminating PINX, and enter the Originating_connection_request state.

10.2.2.1.2 Actions in the Originating_connection_request state

If the protocol entity informs GFT-Control that a CALL PROCEEDING message has been received, GFT-Control shall start timer T310.

If the protocol entity informs GFT-Control that a RELEASE or RELEASE COMPLETE message has been received, GFT-Control shall inform the ASE that the connection has failed, stop timer T310 and enter the Originating_connection_idle state.

If the protocol entity informs GFT-Control that a CONNECT message has been received, GFT-Control shall stop timer T310 and enter the Originating_connection_active state.

If timer T310 expires, GFT-Control shall inform the ASE that connection establishment has failed and request the protocol entity to send a RELEASE message.

10.2.2.1.3 Actions in the Originating_connection_active state

If the protocol entity informs GFT-Control that a RELEASE message has been received, GFT-Control shall inform the ASE that the connection has been released and enter the Originating_connection_idle state.

If a request that the connection be released is received from the ASE, GFT-Control shall request that the protocol entity send a RELEASE message and enter the Originating_connection_idle state.

10.2.2.2 Actions at a Transit PINX

If GFT-Control receives indication from the protocol entity of a received CO-BI SETUP message from the Preceding PINX, it shall examine the contents of the Called party number information element. If the Called party number information element matches that of the Receiving PINX, the PINX shall become a Terminating PINX. Otherwise it shall follow the procedures of the subclauses below.

10.2.2.2.1 Actions in the Transit_connection_idle state

If the contents of the Called party number information element contained in the CO-BI SETUP message is that of another PINX and a connection to that PINX is possible, GFT-Control shall request the protocol entity to send a CO-BI SETUP message on the appropriate inter-PINX link to the Succeeding PINX, associate the incoming and outgoing connections and enter the Transit_connection_request state.

If the contents of the Called party number information element contained in the CO-BI SETUP message is not sufficient to enable routing onto a further inter-PINX link, GFT-Control shall request the protocol entity to release the connection by sending a RELEASE message to the Preceding PINX and remaining in the Transit_connection_idle state.

10.2.2.2.2 Actions in the Transit_connection_request state

If the protocol entity informs GFT-Control that a CALL PROCEEDING message has been received from the Succeeding PINX, GFT-Control shall start timer T310.

When the protocol entity informs GFT-Control of a CONNECT message received from the Succeeding PINX, GFT-Control shall request the protocol entity to send a CONNECT message to the Preceding PINX, stop timer T310 if running and enter the Transit_connection_active state.

When the protocol entity informs GFT-Control that a RELEASE or RELEASE COMPLETE message has been received from the Succeeding PINX, GFT-Control shall request the protocol entity to send a RELEASE message to the Preceding PINX, stop timer T310 if running and enter the Transit_connection_idle state.

When the protocol entity informs GFT-Control that a RELEASE message has been received from the Preceding PINX, GFT-Control shall request the protocol entity to send a RELEASE message to the Succeeding PINX, stop timer T310 if running and enter the Transit_connection_idle state.

If timer T310 expires, GFT-Control shall request the protocol entity to send a RELEASE message to the Preceding PINX and request the protocol entity to send a RELEASE message to the Succeeding PINX.

10.2.2.2.3 Actions in the Transit_connection_active state

If the protocol entity informs GFT-Control of the receipt of a RELEASE message from the Succeeding PINX, GFT-Control shall request the protocol entity to send a RELEASE message to the Preceding PINX and shall enter the Transit_connection_idle state.

If the protocol entity informs GFT-Control of the receipt of a RELEASE message from the Preceding PINX, GFT-Control shall request the protocol entity to send a RELEASE message to the Succeeding PINX and shall enter the Transit_connection_idle state.

10.2.2.3 Actions at a Terminating PINX

10.2.2.3.1 Actions in the Incoming_connection_idle state

If the protocol entity notifies GFT-Control of a received CO-BI SETUP message that is to be terminated on the receiving PINX, and resources for the connection are available, GFT-Control shall request the protocol entity to send a CONNECT message and shall enter the Incoming_connection_active state.

If no resources for the connection are available, GFT-Control shall request the protocol entity to send a RELEASE message and shall remain in the Incoming_connection_idle state.

10.2.2.3.2 Actions in the Incoming_connection_active state

If the protocol entity informs GFT-Control that a RELEASE message has been received from the Preceding PINX, it shall inform the ASE that the connection has been released and enter the Incoming_connection_idle state.

If the ASE requests that the connection be released, GFT-Control shall request that the protocol entity send a RELEASE message and shall enter the Incoming_connection_idle state.

10.3 Remote operations procedures

10.3.1 Introduction

Introduction described in ITU-T Rec. Q.2932.1 clause 9.4.1 shall apply.

10.3.2 Procedures for operations

Procedures for operations described in ITU-T Rec. Q.2932.1 clause 9.4.2 shall apply.

10.3.2.1 Invocation

Invocation described in ITU-T Rec. Q.2932.1 clause 9.4.2.1 shall apply.

10.3.2.2 Return result

Return result described in ITU-T Rec. Q.2932.1 clause 9.4.2.2 shall apply.

10.3.2.3 Return error

Return error described in ITU-T Rec. Q.2932.1 clause 9.4.2.3 shall apply.

10.3.2.4 Reject

Reject described in ITU-T Rec. Q.2932.1 clause 9.4.2.4 shall apply.

10.3.2.5 Formal definition of data types

Formal definition of data types described in ITU-T Rec. Q.2932.1 clause 9.4.2.5 shall apply.

10.4 Notification transport mechanisms

For the CL-BI transport mechanism, the Notification indicator information element may be included in the FACILITY message. The following procedures apply for the bearer related and CO-BI transport mechanisms.

10.4.1 Sending notification information

The transport of notifications shall make use of the call reference of a call/connection or a CO-BI connection. Notifications shall be sent using the Notification indicator information element.

If the delivery of the notification information coincides with the sending of the FACILITY message or any basic call/connection control or CO-BI connecton control message in which the Notification indicator information element is permitted, the notification may be carried in that message. Otherwise, the notification shall be delivered in a NOTIFY message.

However:

- if a SETUP or CO-BI SETUP message has been sent, but no response has been received from the Next PINX; or
- if a SETUP or CO-BI SETUP message has been received from the Preceding PINX, but no response has been sent; or,
- if a clearing message has already been sent to or received from the Next PINX

the notification information shall be discarded.

No state change shall occur on sending a NOTIFY message.

10.4.2 Receiving notification information

On receipt of a Notification indicator information element, in the NOTIFY message or in any other message in which the Notification indicator information element is permitted, it shall be passed to GFT-Control.

No state change shall occur on receipt of a NOTIFY message.

10.5 GFT-Control procedures for notifications

10.5.1 Actions at a PINX which generates notifications

A PINX which wishes to generate a notification shall request the protocol entity to send a Notification indicator information element.

10.5.2 Actions at a receiving PINX

For the CL-BI transport mechanism, the handling of a received Notification indicator information element is outside the scope of this Standard. The following procedures apply for the bearer-related and CO-BI transport mechanisms.

10.5.2.1 Actions at a Transit PINX

If a Transit PINX receives a Notification indicator information element from the Preceding PINX, it shall request the protocol entity to send the Notification indicator information element to the Succeeding PINX.

If a Transit PINX receives a Notification indicator information element from the Succeeding PINX, it shall request the protocol entity to send the Notification indicator information element to the Preceding PINX.

10.5.2.2 Actions at a Receiving End PINX

If an End PINX receives a Notification indicator information element at any time during a call/connection, it shall convey the information it contains to the PISN user if the PISN user's equipment is able to receive such information.

If an End PINX receives a Notification indicator information element at any time during a CO-BI connection, it shall convey the information it contains to the PISN user if there is a PISN user associated with that end of the connection and if the PISN user's equipment is able to receive such information.

NOTE
Further (implementation specific) actions of a PINX receiving a notification (e.g. changing the state of a local non-standard state machine) are not precluded and are beyond the scope of this Standard.

11 Interworking with (narrowband) QSIG

Two means exist for interworking with the N-PISN. In the first the generic functional protocol is fully terminated. In the second, a generic interworking function is provided.

An interworking PINX shall provide the procedures of 11.1 for full termination of the B-QSIG and QSIG protocols. An interworking PINX may also provide the generic interworking procedures of 11.2 on a case by case basis.

When an interworking PINX receives generic functional data from a B-QSIG link and is able to support the optional procedures of 11.2, the decision to pass the generic functional data on to the QSIG link unchanged in accordance with 11.2 shall be based on the contents of the NFE (if any) as shown in table 5

Likewise, when an interworking PINX receives generic functional data from a QSIG link and is able to support the optional procedures of 11.2, the decision to pass the generic functional data on to the B-QSIG link unchanged in accordance with 11.2 shall be based on the contents of the NFE (if any) as shown in table 5

Table 5 - Effect of NFE on handling of generic functional data at a B-QSIG to QSIG interworking PINX

NFE contents	Action
No NFE	Terminate in accordance with 11.1
NFE with destinationEntity value endPINX	Optionally provide generic interworking; otherwise terminate in accordance with 11.1
NFE with destinationEntity value anyTypeOfPINX and no destinationEntityAddress	Optionally provide generic interworking; otherwise terminate in accordance with 11.1
NFE with destinationEntity value anyTypeOfPINX and a destinationEntityAddress	Terminate in accordance with 11.1 if the address matches the interworking PINX's address; otherwise provide generic interworking

For cases where the specified action is "optionally provide generic interworking", generic interworking may be used if the only ROSE APDUs are Invoke APDUs either with operation values that are valid in the other network and which are more appropriately terminated in the other network or with operation values that are unrecognised. If the generic functional data contains ROSE APDUs other than Invoke APDUs or contains Invoke APDUs that are not valid in the other network or are more appropriately terminated at the interworking PINX, full termination in accordance with 11.1 shall be employed.

11.1 Full termination of generic functional protocol

Full termination of generic functional protocol described in ITU-T Rec. Q.2932.1 clause 11.1.1 shall apply.

11.2 Generic interworking function

11.2.1 Architecture

Figure 4 shows the protocol architecture of this interworking mechanism.

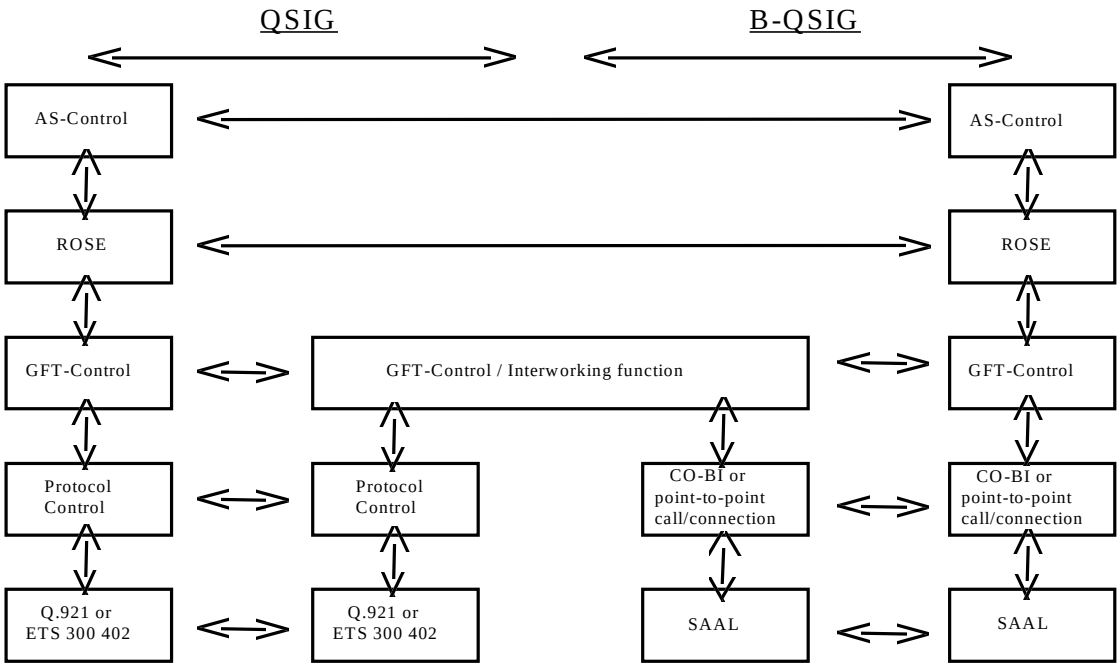


Figure 4 - Generic interworking

For this form of interworking to take place, the supplementary service procedures or other functionality for both QSIG and B-QSIG are identical with the exception of the transport mechanism. The same operation and error values shall be used in both protocols for the same supplementary service or other functionality.

The procedures for interworking the various transport mechanisms are given in the following subclauses.

11.2.2 Bearer-related transport mechanism

All mapping is performed as specified in ISO/IEC 13247 with the addition that the Facility information element is included in all mapped messages.

In addition, for mapping B-QSIG to QSIG the mappings shown in table 6 shall apply.

Table 6 - B-QSIG to QSIG mapping

B-QSIG message		QSIG message
FACILITY	----->	FACILITY

In addition, for mapping QSIG to B-QSIG , the mappings shown in table 7 shall apply.

Table 7 - QSIG to B-QSIG mapping

QSIG message		B-QSIG message
FACILITY	----->	FACILITY

The B-QSIG Facility information element is mapped to the QSIG Facility information element by removing its second octet and adjusting the length indication without causing other changes to the contents.

The QSIG Facility information element is mapped to the B-QSIG Facility information element by inserting the second octet and changing length indication field from one to two octets, adjusting the length accordingly, without causing other changes to the contents. The flag bit in the second octet is set to "0", i.e. the normal error handling procedures as defined in subclause 9.6 of ISO/IEC 13247 apply.

10.2.3 Connection-oriented bearer independent mechanism

For mapping B-QSIG to QSIG , the mappings shown in table 8 shall apply.

Table 8 - B-QSIG to QSIG mapping

B-QSIG message		QSIG message
CO-BI SETUP (NOTE 1)	------>	SETUP
CALL PROCEEDING		not mapped
CONNECT	----->	CONNECT
FACILITY	----->	FACILITY
RELEASE (NOTE 2)	----->	RELEASE
RELEASE COMPLETE	----->	RELEASE (NOTE 3)
NOTIFY (NOTE 4)		not mapped
NOTE 1 – A CALL PROCEEDING message is also returned to the B-QSIG entity by the interworking function.		
NOTE 2 – A RELEASE COMPLETE message is also returned to the B-QSIG entity by the interworking function.		
NOTE 3 – This mapping only occurs if the B-QSIG RELEASE COMPLETE message is the first clearing message.		
NOTE 4 – It is not expected that this message would occur in a B-QSIG to QSIG interworking scenario.		

For mapping QSIG to B-QSIG , the mappings shown in table 9 shall apply.

Table 9 - QSIG to B-QSIG mapping

QSIG message		B-QSIG message
SETUP (NOTE 1)	------>	CO-BI SETUP
CALL PROCEEDING		not mapped
CONNECT (NOTE 2)	------>	CONNECT
CONNECT ACKNOWLEDGE		not mapped
FACILITY	----->	FACILITY
RELEASE (NOTE 3)	----->	RELEASE
RELEASE COMPLETE		RELEASE (NOTE 4)
NOTE 1. A CALL PROCEEDING message is returned to the QSIG entity by the interworking function		
NOTE 2. A CONNECT ACKNOWLEDGE message is returned to the QSIG entity by the interworking function.		
NOTE 3. A RELEASE COMPLETE message is returned to the QSIG entity by the interworking function.		
NOTE 4. This mapping occurs only if the QSIG RELEASE COMPLETE message is the first clearing message.		

For the mappings shown in table 8 and table 9 the following information elements are mapped in either direction:

– Facility information element;
– Called party number information element;
– Calling party number information element

The contents of the following information elements contained in the B-QSIG protocol are discarded:

– Notification indicator information element.

The contents of the following information elements contained in the QSIG protocol are discarded:

– Bearer capability information element;
– Channel identification information element;
– Sending complete information element.

The B-QSIG Facility information element is mapped to the QSIG Facility information element by removing its second octet and adjusting the length indication without causing other changes to the contents.

The QSIG Facility information element is mapped to the B-QSIG Facility information element by inserting the second octet and changing length indication field from one to two octets, adjusting the length accordingly, without causing other changes to the contents. The flag bit in the second octet is set to "0", i.e. the normal error handling procedures as defined in subclause 9.6 of ISO/IEC 13247 shall apply.

Other information elements are mapped as defined in ISO/IEC 13247 annex B.

12 Parameter values

12.1 Connection-oriented bearer-independent transport

Protocol timer values specified in ITU-T Rec. Q.2932.1 clause 12.1 shall apply.

The GFT-Control timer value T310 specified in table 19 in ITU-T Rec. Q.2932.1 clause 12.1 shall apply, with the same conventions and tolerances as for table 18. This timer shall be mandatory at an Originating PINX and optional at a Transit PINX.

13 Dynamic description (SDLs)

Dynamic Description (SDL) described in ITU-T Rec. Q.2932.1 clause 13 shall apply.

13.1 Block overview diagram

Block overview diagram described in ITU-T Rec. Q.2932.1 clause 13.1 shall apply with the following modification.

– The processes Q.2931_U and Q.2931_N shall be replaced by a single process "ISO/IEC 13247 protocol control". This shall comprise the Protocol Control process specified in ISO/IEC 13247 enhanced as specified in 13.2.1.

13.2 Component transport mechanisms

13.2.1 Bearer-related transport mechanism

For bearer-related transport, the Protocol Control SDL of ISO/IEC 13247 shall be enhanced as specified in figure 1-10 of Q.2932.1 with the following modification.

– All states are ISO/IEC 13247 protocol control states.

13.2.2 Connection-oriented bearer-independent transport mechanism

Connection-oriented bearer-independent transport mechanism described in ITU-T Rec. Q.2932.1 clause 13.2.2 shall apply with the following modification.

– In state 0 (Null), on receipt of a CO-BI SETUP message, a CALL PROCEEDING output symbol (to the right) shall be shown before the CO-BI-Setup.ind. output symbol.

– In state 6 (Call Present), the branch beginning with the input symbol CO-BI Proceed.req shall not apply.

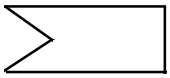
13.2.3 Connectionless bearer-independent transport mechanism

Connectionless bearer-independent transport mechanism described in ITU-T Rec. Q.2932.1 clause 13.2.3 shall apply.

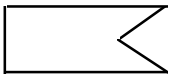
13.3 GFT-Control

The SDL diagram for APDU aspects of GFT-Control is shown in following flows.

Legend

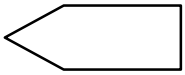


Signal from ROSE to GFT-control (note)

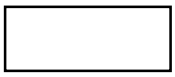


Signal from a Protocol entity to GFT-control (note)

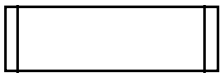
Signal from GFT-control to a Protocol entity



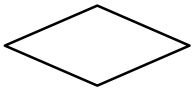
Signal from GFT-control to ROSE (note)



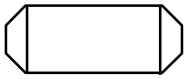
Task symbol



Calling macro



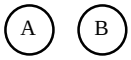
Decision symbol



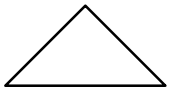
Start of macro

Connector number i

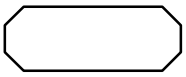
Exit from macro



Connector(s)



Implementation symbol



State symbol

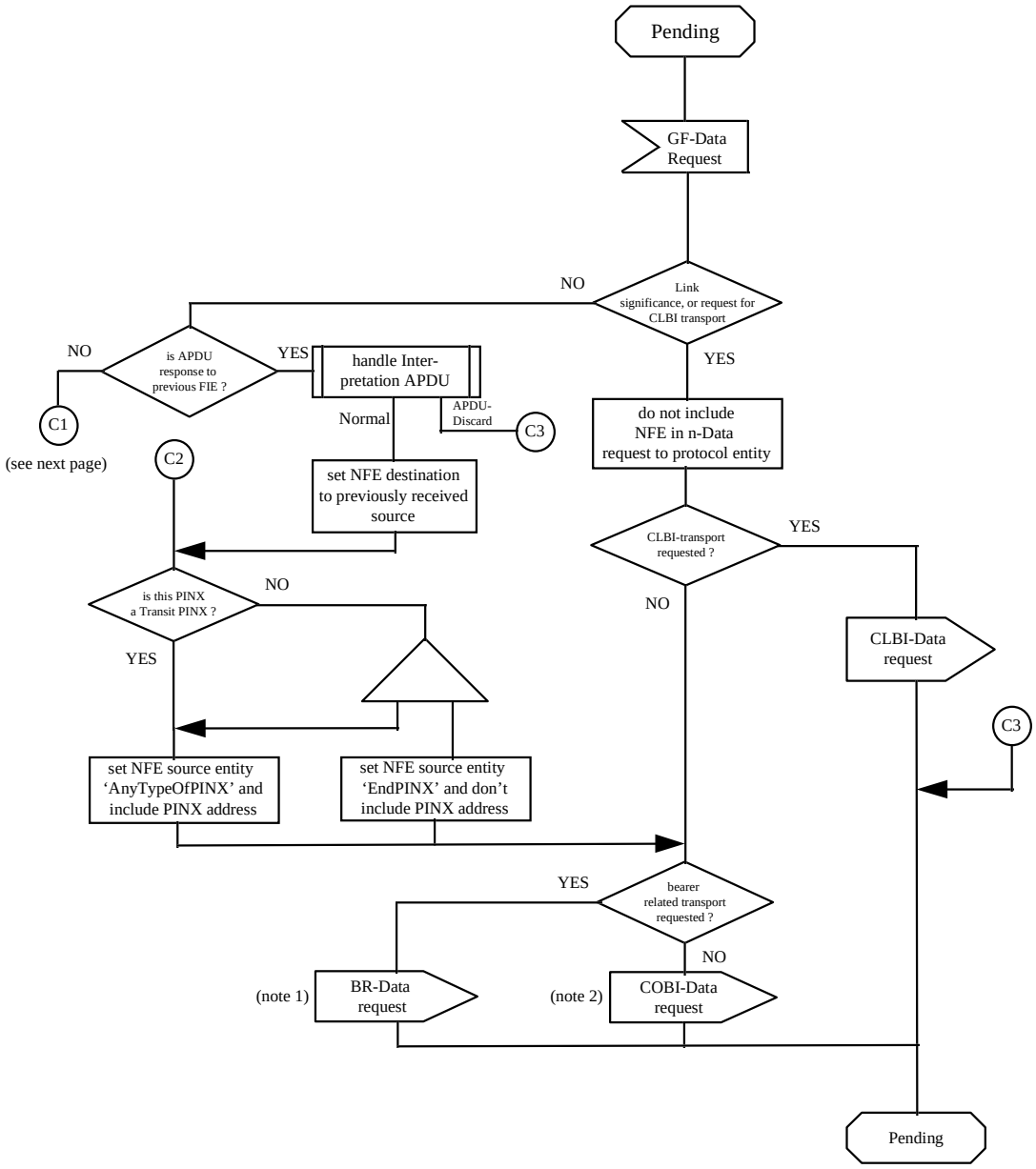
Save

FIE = Facility Information Element

Note: These signals are sent via the coordination function

:

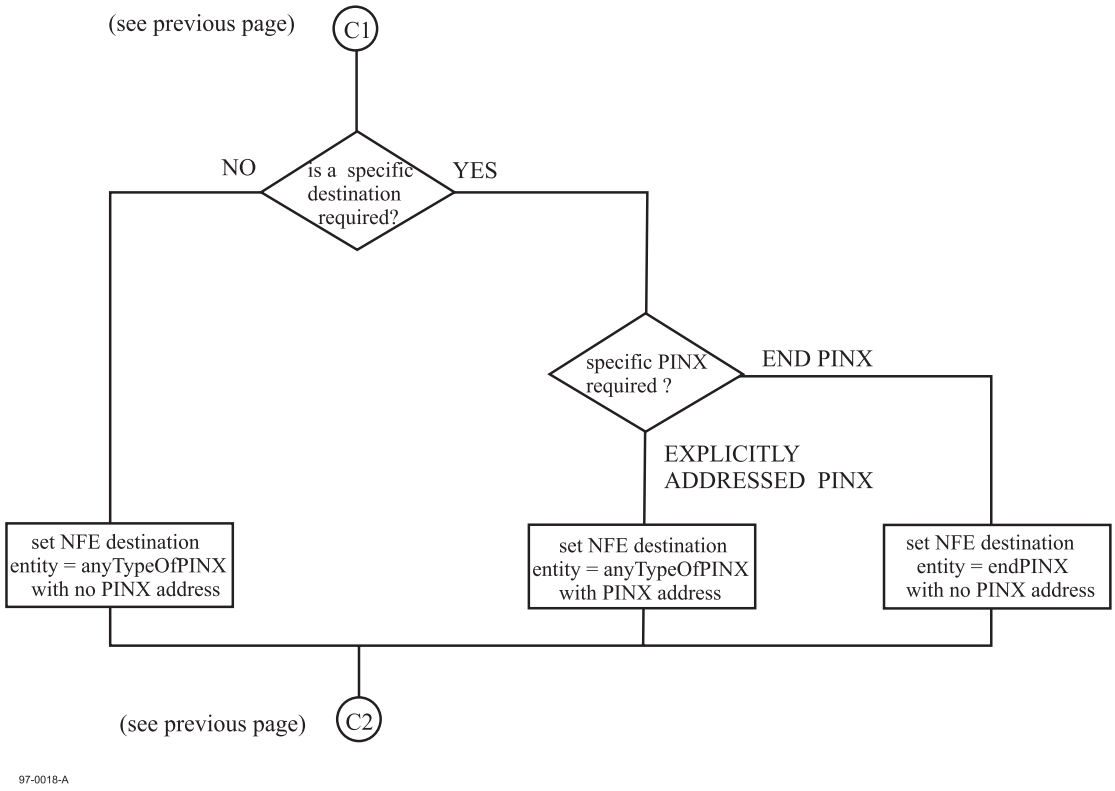
SDL for GFT-control for the transport of APDUs



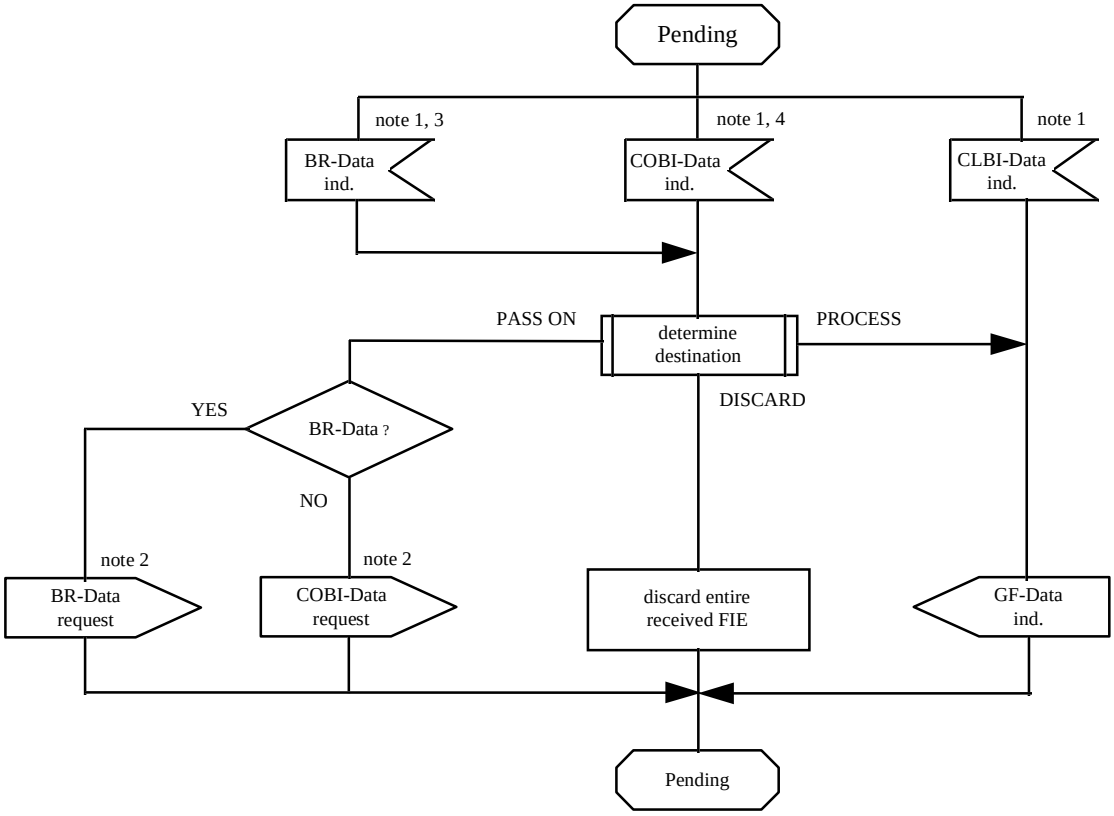
Note 1: Such information can be synchronized with bearer-realated messages within the co-ordination process. Such synchronization is implementation dependent, and therefore not shown in this SDL.

Note 2: If appropriate, data can also be sent in the COBI-setup request, COBI-setup response and COBI-release reqeuest primitives. It is an implementation dependent matter and is outside the scope of this standard as to when the COBI process is established.

SDL for GFT-control for the transport of APDUs



SDL for GFT-control for the transport of APDUs



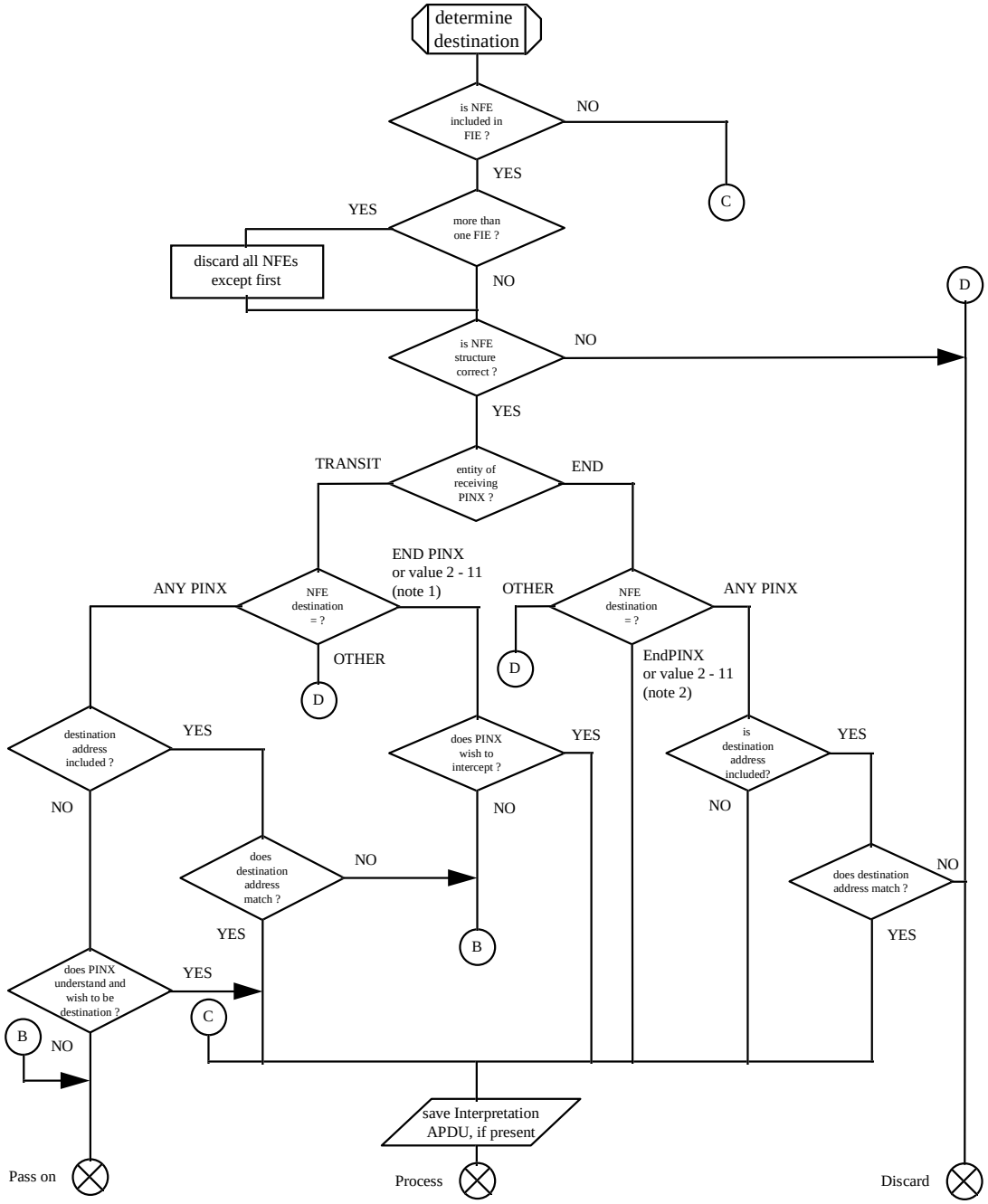
Note 1: This primitive indicates that the Protocol entity has received a Facility information element from an Adjacent PINX.

Note 2: This primitive to the Protocol entity causes a Facility information element to be sent to the Next PINX in the direction of the Destination PINX.

Note 3: Such information can be synchronized with bearer-realated messages within the coordination process. The mechanism for separating this information is performed by the coordination process.

Note 4: Such information can also appear in the COBI-setup indication, COBI-setup confirm and COBI-release indication primitives. As the time of establishment/release of the COBI transfer mechanism is outside the scope of this standard, this SDL does not provide this in detail.

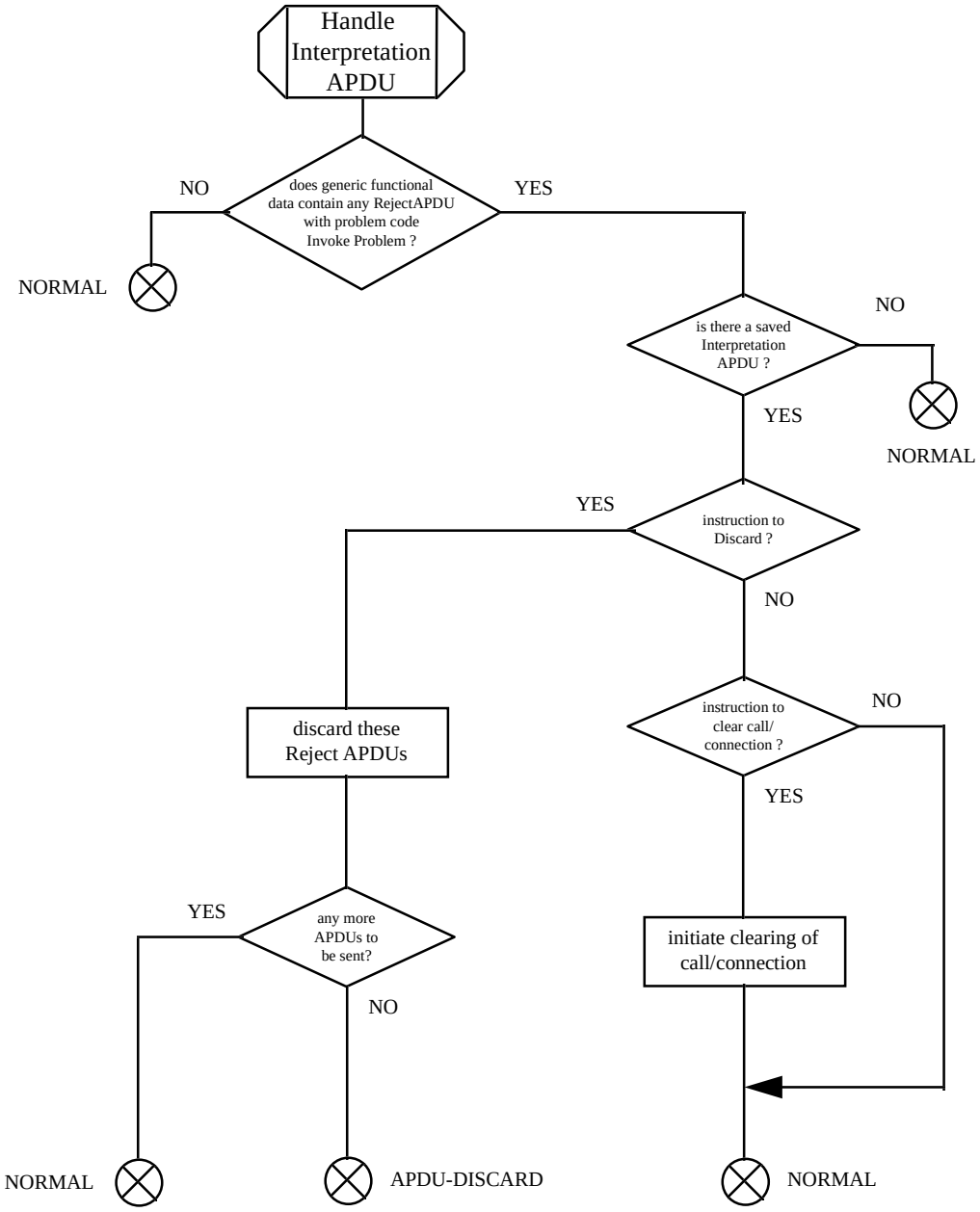
SDL for GFT-control for the transport of APDUs



Note 1: Values 2 to 11 are reserved for future use (see subclause 10.2.1.2.2)

Note 2: Values 2 to 11 are reserved for future use (see subclause 10.2.1.2.1)

SDL for GFT-control for the transport of APDUs



14 Manufacturer Specific Information (MSI)

B-QSIG permits the inclusion in messages of non-standardised information which is specific to a particular design of PINX or a particular network etc. This information is known as Manufacturer Specific Information (MSI).

Manufacturer specific information may exist in the PISN a result of the following:

– manufacturer specific Supplementary services;
– manufacturer specific extensions to Standard Supplementary services; or
– manufacturer specific notifications

In all these cases, any information which is manufacturer specific shall be encoded in such a way that it can be uniquely identified. Apart from the use of information elements belonging to codesets 6 or 7, as described in ISO/IEC 13247 for conveyance of MSI to an Adjacent PINX, any manufacturer specific information generated by a PINX conforming to this International Standard shall be encoded in conformance with the contents of this clause.

14.1 Manufacturer specific operations and errors

Manufacturer specific operations and errors shall conform to the encoding and transport rules defined for standardised operations and errors in other clauses of this Standard, but in addition shall make use of operation values and error values which are unique to that manufacturer - i.e. of type OBJECT IDENTIFIER. Examples of how manufacturer specific operations may be encoded are shown in annex H.

14.2 Manufacturer specific additions to standardised operations and error

As an alternative to the definition of a manufacturer specific operation or error, a manufacturer may wish to use an enchanced form of a standardised operation or error.

NOTE
This may be used , for example, to include additional parameters which are manufacturer specific as part of the standard service (e.g. information describing the detailed location of a party involved in the service).

To allow for this possibility Standards for Supplementary services or additional basic call capabilities will include ‘placeholders’ for manufacturer specific extensions. Each placeholder will be an optional CHOICE construct containing an element of type Extension or a sequence of elements of type Extension (as defined in B-8 of annex B using ASN.1 as specified in ITU-T recommendation X.208 or C-8 of annex C using ASN.1 as specified in ITU-T recommendation X.680) within the argument or result of an operation or within the parameter of an error. This placeholder may be included in the ROSE APDU if MSI is to be conveyed. An element of type Extension shall contain an element of type OBJECT IDENTIFIER to uniquely identify the MSI.

If the Destination PINX identifies an element of type Extension or sequence of elements of type Extension in a standardised operation, when processing the contents of a received Facility information element in accordance with the relevant Supplementary service standard or additional basic call capability, it shall act on an element of type Extension only if it recognises the value in the element of type OBJECT IDENTIFIER, (see B-8 of annex B using ASN.1 as specified in ITU-T recommendation X.208 or the EXTENSION object class specified in C-8 of annex C using ASN.1 as specified in ITU-T recommendation X.680). Otherwise the entire element of type Extension shall be discarded. In the case of a sequence of elements of type Extension (i.e. where multiple extensions to the service are defined) the PINX shall consider each element of type Extension separately - that is, only those elements of type Extension containing an unrecognised value in the element of type OBJECT IDENTIFIER shall be discarded.

A manufacturer specific extension may be defined using the EXTENSION macro specified in B-8 of annex B using ASN.1 as specified in ITU-T recommendation X.208 or the EXTENSION object class specified in C-8 of annex C using ASN.1 as specified in ITU-T recommendation X.680.

An example of the use of the Extension type is shown in annex H.

14.3 Manufacturer specific notifications

Manufacturer specific notifications may occur in the PISN as part of manufacturer specific Supplementary services or as additions to standardised Supplementary services. If provided, they shall be encoded and transported across the PISN in a accordance with the rules for standardised notification (see 10.1.1, 9.1 and 9.2.2.3).

Manufacturer specific notification shall be conveyed using ASN.1 type NotificationDataStructure in octet 5.1 of the Notification indicator information element, as specified in 9.2.2.3.

Manufacturer specific notifications shall not make use of the notification description field (octet 5) of the Notification indicator information element, other than to include the ‘discriminator for notification extension’ codepoint (see 9.2.2.3).

Annex A

(normative)

Protocol Implementation Conformance Statement (PICS) proforma

A.1 Introduction

A.1.1 Basic reference documents for PICS proforma specifications

General rules for the specification of PICS proforma are provided by ISO/IEC 9646-1. Detailed guidance for the specification of PICS proforma is provided by ISO/IEC 9646-7; in particular the structure of a PICS proforma, the questions to be asked, the syntax and notation to be used and the semantics of the questions and expected answers.

For a PICS proforma, specific acronyms and terms are used as defined in ISO/IEC 9646-1 or ISO/IEC 9646-7, e.g.:

- ICS	Implementation Conformance Statement
- ICS proforma	Implementation Conformance Statement proforma
- ICS (proforma) item	A row in an ICS (proforma) table
- PICS	Protocol ICS
- PICS proforma	Protocol ICS proforma
- status (value)	An allowed entry in the status column for an item in an ICS proforma table
- (support) answer	An allowed entry in the support or supported values columns for an item in an ICS question

A.1.2 Copyright Information

Users of this specification may freely reproduce the PICS proforma of this annex A so that it can be used for its intended purpose and may further publish the completed PICS.

A.1.3 Structure of this PICS proforma

This PICS proforma is subdivided into (sub-)clauses as follows:

- Instructions (A.2)
- Purpose of a PICS proforma (A.2.1)
- Instructions for completing the PICS proforma (A.2.2)
- Additional Information (A.2.3)
- Exception Information (A.2.4)
- Legend for the columns of the PICS proforma tables (A.2.5)
- Legend for further indications of the PICS proforma tables (A.2.6)
- Identification of the implementation (A.3), including:
- Identification of the protocol for which this PICS applies (A.3.7)
- Global statement of conformance (A.4)
- Roles (A.5)
- Major capabilities (A.6)
- Subsidiary capabilities (A.7)
- Protocol data units (A.8)
- Protocol data unit parameters (A.9)

A.2 Instructions

A.2.1 Purpose of a PICS proforma

To evaluate conformance of a particular implementation, it is necessary to have a statement of which capabilities and options have been implemented for a given OSI specification. Such a statement is called an Implementation Conformance Statement (ICS).

For protocol specifications, this statement is called "Protocol Implementation Conformance Statement" (PICS). For the provision of this statement, a fixed format questionnaire called PICS proforma has to be used. A completed PICS proforma is the PICS for the implementation in question. It is an ICS (as defined in ISO/IEC 9646-7) for an implementation or system which claims to conform to a given specification.

The PICS can have a number of uses, including:

- by the protocol implementor, as a check list for implementations to reduce the risk of unintended non-conformance, e.g. through oversight;
- by the supplier and acquirer, or potential acquirer, of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the Standard's PICS proforma;
- by the user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation - while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICS
- by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

The PICS proforma of this annex therefore reflect a compromise between these different requirements.

A.2.2 Instructions for completing the PICS proforma

The supplier of a protocol implementation which is claimed to conform to this Standard shall complete the following Protocol Implementation Conformance Statement (PICS) proforma.

The PICS proforma is a fixed format questionnaire. The supplier of the implementation shall complete this questionnaire, in particular identify the implementation, complete the global statement of conformance, and providing the answers in the rows of the tables in clauses A.5 - A.9. The structure of the tables is explained in subclauses A.2.5 and A.2.6. For each row in each table, the supplier shall enter an explicit answer (i.e. by ticking the appropriate "yes", "no", or "N/A" in each of the support column boxes provided. Where a support column box is left blank, or where it is marked "N/A" without any tick box, no answer is required. If a "prerequisite line" (see A.2.6 below) is used after a subclause heading or table title, and its predicate is false, no answer is required for the whole sublause or table, respectively.

A supplier may also provide - or be required to provide - further information, categorised as either Additional Information or Exception Information. When present, each kind of further information is to be provided in a further subclause of items labelled

"a.<i>" for additional information,

"x.<i>" for exceptional information

for cross-referencing purposes, where <i> is any unambiguous identification for the item (e.g., simply a numeral); there are no other restrictions on its format and presentation.

A.2.3 Additional Information

Items of Additional Information allow a supplier to provide further information intended to assist the interpretation of the PICS. It is not intended or expected that a large quantity will be supplied, and a PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception information.

A.2.4 Exception Information

It may occasionally happen that a supplier will wish to answer an item with mandatory or prohibited status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer will be found in the Support column for this. Instead, the supplier is required to write into the support column an x.<i> reference to an item of Exception Information, and to provide the appropriate rationale in the Exception item itself.

An implementation for which an Exception item is required in this way does not conform to this Standard; and the answer to the global statement of conformance (see A.4) cannot be "yes". A possible reason for the situation described above is that a defect in the Standards has been reported, a correction for which is expected to change the requirement not met by the implementation.

A.2.5 Legend for the columns of the PICS proforma tables

The questionnaire in clauses A.5-A.9 is structured as a set of tables in accordance with the guidelines presented in ISO/IEC 9646-7. The columns of the tables shall be interpreted as follows:

"Item"

The item column contains a unique reference (a mnemonic plus a number) for each item within the PICS proforma. Items need not always be numbered sequentially.

"Item Description"

The item description column contains a brief summary of the static requirement for which a support answer is required. This may be done by a question or a reference to a specific feature.

"Conditions for Status"

The conditions for status column contains a specification, if appropriate, of the predicate upon which a conditional status is based. The indication of an item reference in this column indicates a simple-predicate condition (support of this item is dependent on the support marked for the referenced item).

Within the "conditions for status" column, the logical symbol "]" is used to indicate a logical negation ("NOT").

"Status"

The following notations, as defined in ISO/IEC 9646-7, are used for the status column:

I Irrelevant or out-of-scope - this capability is outside the scope of the standard to which this PICS proforma applies and is not subject to conformance testing in this context.

M Mandatory - the support of this capability is required for conformance to the standard

N/A Not Applicable - in the given context, it is impossible to use the capability. No answer in the support column is required.

O Optional - the capability is not required for conformance to the protocol and may be supported or not. However, if the capability is implemented, it is required to conform to the protocol specifications.

O.<n> Qualified optional - in this case, <n> is an integer that identifies a unique group of related optional items. If no additional qualification is indicated, the support of at least one of the optional items is required for conformance to the standard. Otherwise, the qualification and logic of the selection among the optional items is defined below the table explicitly.

X eXcluded or prohibited - there is a requirement not to use this capability in a given context.

"Reference"

Except where explicitly stated, the reference column refers to the appropriate subclause(s) of this Standard describing the particular item. The reference merely indicates the place(s) where the core of a description of an item can be found; additional information on this item may be contained in other parts of this Standard, and has to be taken into account when making a statement about the conformance to that particular item.

"Support "

In the support column, the supplier of the implementation shall enter an explicit answer. The following notation is used:

[ ] Yes [No] Tick "yes", if item is supported; tick "No", if item is not supported.

[ ] N/A Tick "N/A", if the item is "not applicable".

In specific cases, the indication of explicit values may be requested. Where a support column box is left blank, or where it is marked "N/A" without any tick box, no answer is required.

A.2.6 Legend for further indications of the PICS proforma tables

In addition to the columns of a table, the following information may be indicated:

"Prerequisite line"

A prerequisite line after a subclause heading or table title indicates that the whole subclause or the whole table is not required to be completed if the predicate is false. The prerequisite line takes the form:

Prerequisite:<predicate>.

"Qualification"

At the end of a table, a detailed qualification for a group of optional items may be indicated, as specified in the description of the status "qualified optional" in subclause A.2.5.

"Comments"

This box at the end of a table allows a supplier to enter any comments to that table. Comments may also be provided separately (without using this box).

A.3 Identification of the implementation

Identification of the implementation and the system in which it resides should be filled in to provide as much detail as possible regarding version numbers and configuration options.

The implementation about which this PICS proforma asks questions corresponds to a B-QSIG GF implementation at the Q reference point.

Configuration options outlined in B-QSIG GF have been incorporated into this PICS proforma. They are referred to by qualified options or prerequisite lines, in order to reflect that an implementation only needs to provide the addressed functions at an interface, if it is configured accordingly (e.g. an implementation only needs to provide gateway call handling functions, if it is configured to act as gateway PINX at an interface).

The contact person indicated (see A.3.6) should be able to answer queries regarding information supplied in the PICS.

As specified in clause 5 of ISO/IEC 9646-7, it is required for all implementations to at least provide the identification of the implementation (A.3.2), product supplier information (A.3.4), identification of a contact person (A.3.6), and detailed identification of the protocol for which the PICS applies (A.3.7). Identification of the system in which the implementation resides (A.3.3) is recommended in order to facilitate full identification of the system, and avoid possible problems during conformance testing. The client information (A.3.5) only needs to be filled in if it is relevant and different from the product supplier information.

A.3.1 Date of statement

_____

A.3.2 Identification of the implementation

The terms "name" and "version" should be interpreted appropriately to correspond with a suppliers terminology (e.g. Type, Series, Model).

Name of the implementation:

_____

_____

Implementation version:

_____

_____

A.3.3 Identification of the system in which it resides

Name of the system:

_____

_____

Hardware configuration:

_____

_____

_____

Operating system:

_____

_____

A.3.4 Product supplier

Name:

_____

Address:

_____

_____

_____

Telephone number:

_____

Facsimile number:

_____

E-Mail address:

_____

Additional information:

_____

_____

_____

A.3.5 Client

Name:

_____

Address:

_____

_____

_____

Telephone number:

_____

Facsimile number:

_____

E-Mail address:

_____

Additional information:

_____

_____

_____

A.3.6 PICS contact person

Name:

_____

Address:

_____

_____

_____

Telephone number:

_____

Facsimile number:

_____

E-Mail address:

_____

Additional information:

_____

_____

_____

A.3.7 Protocol for which this PICS applies

Protocol:

B-QSIG GF - B-PISN inter-exchange signalling protocol - Generic Functional Protocol

Protocol Version - please identify the standards document unambiguously, including e.g. reference number (e.g. ECMA-<nnn>), edition number and publication date:

_____

_____

Corrigenda Implemented (if applicable):

_____

_____

Addenda Implemented (if applicable):

_____

_____

Amendments Implemented (if applicable):

_____

A.4 Global Statement of Conformance

Does the implementation described in this PICS meet all the mandatory requirements of the referenced standard:

[ ] Yes

[ ] No

Note: Answering "No" to this question indicates non-conformance to the protocol specification. In this case, an explanation shall be given of the nature of non-conformance either below or on a separate sheet of paper. Further the instructions outlined in subclause A.2.5 ("Exception Information") shall be followed when completing the PICS proforma tables.

Nature of non-conformance (if applicable):

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

A.5 Roles

Table A.1 - Type of implementation

Item	Major role: Does the implementation...	Conditions for status	Status	Reference	Support
	Type of implementation				
R 1	not used				
R 2.1	support transit PINX?		O.1		[ ]Yes [ ]No
R 2.2	support originating PINX?		O.1		[ ]Yes [ ]No
R 2.3	support terminating PINX?		O.1		[ ]Yes [ ]No
R 2.4	support incoming gateway PINX?		O.1		[ ]Yes [ ]No
R 2.5	support outgoing gateway PINX?		O.1		[ ]Yes [ ]No
R 3	not used				
R 4	not used				
R 5.1	support the functions of an initiating entity?		O.2		[ ]Yes [ ]No
R 5.2	support the functions of a responding entity?		O.2		[ ]Yes [ ]No
O.1	Support of at least one of these options is required.				
O.2	Support of at least one of these options is required.				
Comments:					

A.6 Major capabilities

Table A.2 - Major capabilities

Item	Major capability: Does the implementation...	Conditions for status	Status	Reference	Support
transport mechanisms					
MC 2.1	support bearer related transport mechanism?		O.3	10.1.1	[ ]Yes [ ]No
MC 2.2	support bearer independent transport mechanism?		O.3	10.1.2	[ ]Yes [ ]No
MC 2.3	support (bearer independent) connection-oriented transport mechanism?	MC 2.2 NOT MC 2.2	O.4 N/A	10.1.3	[ ]Yes [ ]No [ ]N/A
MC 2.4	support (bearer independent) connectionless transport mechanism?	MC 2.2 NOT MC 2.2	O.4 N/A	10.1.4	[ ]Yes [ ]No [ ]N/A
notifications					
MC 3	support transport of notifications?		M	10.4	[ ]Yes [ ]No
MC 3.1	support transport of bearer-related notifications?	MC 2.1 NOT MC 2.1	M N/A	10.4	[ ]Yes [ ]No [ ] N/A
MC 3.2	support transport of bearer-independent notifications?	MC 2.2 NOT MC 2.2	M N/A	10.4	[ ]Yes [ ]No [ ] N/A
MC 3.3	support transport of bearer-independent connection-oriented notifications?	MC 2.3 NOT MC 2.3	M N/A	10.4	[ ]Yes [ ]No [ ] N/A
MC 3.4	support transport of bearer-independent connectionless notifications?	MC 2.4 NOT MC 2.4	M N/A	10.4	[ ]Yes [ ]No [ ] N/A
GFT-control					
MC 4	provide GFT-control as a source PINX?		M	10.2.1.1	[ ]Yes [ ]No
MC 5	provide GFT-control as a receiving PINX?		M	10.2.1.2	[ ]Yes [ ]No
MC 6	act as a destination PINX?		M	10.2.1.3	[ ]Yes [ ]No
MC 7	interwork with (narrowband) QSIG?	R 2.4 OR R 2.5 NOT (R 2.4 OR R 2.5)	O N/A	11	[ ]Yes [ ]No [ ]N/A
MC 7.1	provide interworking by full termination of the generic functional protocol?	MC 7 NOT MC 7	M N/A	11.1	[ ]Yes [ ]No
MC 7.2	provide interworking by generic interworking function?	MC 7 NOT MC 7	O N/A	11.2	[ ]Yes [ ]No
MC 8	support transport of manufacturer specific operations and errors?		M	14.1	[ ]Yes [ ]No
MC 9	support transport of manufacturer specific additions to standardised operations and errors?		M	14.2	[ ]Yes [ ]No
MC 10	support transport of manufacturer specific notification?		M	14.3	[ ]Yes [ ]No
O.3	Support of at least one of these options is required.				
O.4	Support of at least one of these options is required.				
Comments:					

A.7 Subsidiary capabilities

Table A.3 - Subsidiary capabilities

Item	Capability: Does the implementation...	Conditions for status	Status	Reference	Support
SC 3	**Notification procedures**				
SC 3.1	support the transport of simple notifications?		M	9.2.2.3	[ ]Yes [ ]No
SC 3.2	support the transport of ASN.1 encoded notification information?		M	9.2.2.3, Annex B.7 and annex C.7	[ ]Yes [ ]No
Comments:					

A.8 Protocol data units

Table A.4 - Messages received

Item	Message: Does the implementation support the interpretation of...	Conditions for status	Status	Reference	Support
MR 1	CO-BI SETUP?	MC 2.3 NOT MC 2.3	M N/A	9.1.3.2	[ ]Yes [ ]No [ ]N/A
MR 2	FACILITY?		M	9.1.1.1, 9.1.2.1, 9.1.3.4	[ ]Yes [ ]No
NOTE:	These messages are additional to those required for support of basic call/connection (ISO/IEC 13247).				
Comments:					

Table A.5 - Messages transmitted

Item	Message: Does the implementation support the inclusion of...	Conditions for status	Status	Reference	Support
MT 5	CO-BI SETUP?	MC 2.3 NOT MC 2.3	M N/A	9.1.3.2	[ ]Yes [ ]No [ ]N/A
MT 1	FACILITY?		M	9.1.1.1, 9.1.2.1, 9.1.3.4	[ ]Yes [ ]No [ ]N/A
Comments:					

A.9 Protocol data unit parameters

A.9.1 Bearer-related transport mechanism

A.9.1.1 Protocol data unit parameters received

Table A.6 - ALERTING PDU parameters received

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	ALERTING PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 1.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 1.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.7 - CONNECT PDU parameters received

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 2.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 2.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.8 - FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 3.1	Protocol discriminator?	MC 2.1 NOT MC 2.1	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 3.2	Call reference?	MC 2.1 NOT MC 2.1	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 3.3	Message type?	MC 2.1 NOT MC 2.1	M N/A	[2] 8.4	[ ]Yes [ ]No [ ]N/A
IER 3.4	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 3.5	Notification indicator?	MC 2.1 AND MC 3.1 NOT (MC 2.1 AND MC 3.1)	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.9 - PROGRESS PDU parameters received

Item	PROGRESS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 4.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 4.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.10 - RELEASE PDU parameters received

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 5.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 5.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE:	These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).				
Comments:					

Table A.11 - RELEASE COMPLETE PDU parameters received

Item	RELEASE COMPLETE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 6.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 6.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE:	These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).				
Comments:					

Table A.12 - SETUP PDU parameters received

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	SETUP PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 7.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 7.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE:	These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).				
Comments:					

A.9.1.2 Protocol data unit parameters transmitted

Table A.13 - ALERTING PDU parameters transmitted

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	ALERTING PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 1.1	Facility?	MC 2.1 AND R 2.1 MC 2.1 AND NOT R 2.1 NOT MC 2.1	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 1.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.14 - CONNECT PDU parameters transmitted

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 2.1	Facility?	MC 2.1 AND R 2.1 MC 2.1 AND NOT R 2.1 NOT MC 2.1	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 2.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.15 - FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 3.1	Protocol discriminator?	MC 2.1 NOT MC 2.1	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 3.2	Call reference?	MC 2.1 NOT MC 2.1	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 3.3	Message type?	MC 2.1 NOT MC 2.1	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 3.4	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 3.5	Notification indicator?	(MC 2.1 AND MC 3.1) AND R 2.1 (MC 2.1 AND MC 3.1) AND NOT R 2.1 NOT (MC2.1 AND MC 3.1)	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.16 - PROGRESS PDU parameters transmitted

Item	PROGRESS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 4.1	Facility?	MC 2.1 AND R 2.1 MC 2.1 AND NOT R 2.1 NOT MC 2.1	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 4.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.17 - RELEASE PDU parameters transmitted

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 5.1	Facility?	MC 2.1 AND R 2.1 MC 2.1 AND NOT R 2.1 NOT MC 2.1	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 5.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.18 - RELEASE COMPLETE PDU parameters transmitted

Item	RELEASE COMPLETE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 6.1	Facility?	MC 2.1 NOT MC 2.1	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 6.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

Table A.19 - SETUP PDU parameters transmitted

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	SETUP PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 7.1	Facility?	MC 2.1 AND R 2.1 MC 2.1 AND NOT R 2.1 NOT MC 2.1	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 7.2	Notification indicator?	MC 3.1 AND R 2.1 MC 3.1 AND NOT R 2.1 NOT MC 3.1	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE: These parameters are additional to those required for support of basic call/connection (ISO/IEC 13247).					
Comments:					

A.9.2 Connection-oriented bearer-independent transport mechanism

A.9.2.1 Protocol data unit parameters received

Table A.20 - CALL PROCEEDING PDU parameters received

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 8.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 8.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 8.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.21 - CO-BI SETUP PDU parameters received

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 9.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 9.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 9.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 9.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 9.5	Called party number	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IER 9.6	Calling party number	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IER 9.7	Notification indicator?	MC 2.3 AND MC 3.3 NOT (MC 2.3 AND MC 3.3)	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.22 - CONNECT PDU parameters received

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 10.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 10.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 10.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 10.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 10.5	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.23 - FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 11.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 11.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 11.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 11.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 11.5	Notification indicator?	MC 2.3 AND MC 3.3 NOT (MC 2.3 AND MC 3.3)	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.24 - NOTIFY PDU parameters received

Item	NOTIFY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 12.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 12.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 12.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 12.4	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.25 - RELEASE PDU parameters received

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 13.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 13.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 13.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 13.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 13.5	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.26 - RELEASE COMPLETE PDU parameters received

Item	RELEASE COMPLETE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 14.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 14.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 14.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 14.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 14.5	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.27 - STATUS PDU parameters received

Item	STATUS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 15.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 15.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 15.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 15.4	Call state?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IER 15.5	Cause?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.28 - STATUS ENQUIRY PDU parameters received

Item	STATUS ENQUIRY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 16.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 16.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 16.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
Comments:					

A.9.2.2 Protocol data unit parameters transmitted

Table A.29 - CALL PROCEEDING PDU parameters transmitted

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 8.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 8.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 8.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.30 - CO-BI SETUP PDU parameters transmitted

Prerequisite: R 2.1 or R 2.2 or R 2.4

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 9.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 9.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 9.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 9.4	Facility?	MC 2.3 AND R 2.1 MC 2.3 AND NOT R 2.1 NOT MC 2.3	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 9.5	Called party number	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IET 9.6	Calling party number	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IET 9.7	Notification indicator?	(MC 2.3 AND MC 3.3) AND R 2.1 (MC 2.3 AND MC 3.3) AND NOT R 2.1 NOT (MC 2.3 AND MC 3.3)	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.31 - CONNECT PDU parameters transmitted

Prerequisite: R 2.1 or R 2.3 or R 2.5

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 10.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 10.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 10.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 10.4	Facility?	MC 2.3 AND R 2.1 MC 2.3 AND NOT R 2.1 NOT MC 2.3	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 10.5	Notification indicator?	MC 3.3 AND R 2.1 MC 3.3 AND NOT R 2.1 NOT MC 3.3	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.32 - FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 11.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 11.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 11.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 11.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 11.5	Notification indicator?	(MC 2.3 AND MC 3.3) AND R 2.1 (MC 2.3 AND MC 3.3) AND NOT R 2.1 NOT (MC 2.3 AND MC 3.3)	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.33 - NOTIFY PDU parameters transmitted

Item	NOTIFY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 12.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 12.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 12.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 12.4	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE: For the condition to transmit a NOTIFY message see ISO/IEC 13247.					
Comments:					

Table A.34 - RELEASE PDU parameters transmitted

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 13.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 13.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 13.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 13.4	Facility?	MC 2.3 AND R 2.1 MC 2.3 AND NOT R 2.1 NOT MC 2.3	M O N/A	9.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 13.5	Notification indicator?	MC 3.3 AND R 2.1 MC 3.3 AND NOT R 2.1 NOT MC 3.3	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.35 - RELEASE COMPLETE PDU parameters transmitted

Item	RELEASE COMPLETE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 14.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 14.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 14.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 14.4	Facility?	MC 2.3 NOT MC 2.3	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 14.5	Notification indicator?	MC 3.3 AND R 2.1 MC 3.3 AND NOT R 2.1 NOT MC 3.3	M O N/A	9.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table A.36 - STATUS PDU parameters transmitted

Item	STATUS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 15.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 15.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 15.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 15.4	Call state?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
IET 15.5	Cause?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.5	[ ]Yes [ ]No [ ]N/A
Comments:					

Table A.37 - STATUS ENQUIRY PDU parameters transmitted

Item	STATUS ENQUIRY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 16.1	Protocol discriminator?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 16.2	Call reference?	MC 2.3 NOT MC 2.3	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 16.3	Message type?	MC 2.3 NOT MC 2.3	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
NOTE: For the condition to transmit a STATUS ENQUIRY message see ISO/IEC 13247.					
Comments:					

A.9.3 Connectionless bearer-independent transport mechanism

A.9.3.1 Protocol data unit parameters received

Table A.38 - FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 17.1	Protocol discriminator?	MC 2.4 NOT MC 2.4	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IER 17.2	Call reference?	MC 2.4 NOT MC 2.4	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IER 17.3	Message type?	MC 2.4 NOT MC 2.4	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IER 17.4	Facility?	MC 2.4 NOT MC 2.4	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 17.5	Notification indicator?	MC 3.4 NOT MC 3.4	M N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

A.9.3.2 Protocol data unit parameters transmitted

Table A.39 - FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 17.1	Protocol discriminator?	MC 2.4 NOT MC 2.4	M N/A	[2] 8.2	[ ]Yes [ ]No [ ]N/A
IET 17.2	Call reference?	MC 2.4 NOT MC 2.4	M N/A	[2] 8.3	[ ]Yes [ ]No [ ]N/A
IET 17.3	Message type?	MC 2.4 NOT MC 2.4	M N/A	9.2.1	[ ]Yes [ ]No [ ]N/A
IET 17.4	Facility?	MC 2.4 NOT MC 2.4	M N/A	9.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 17.5	Notification indicator?	MC 3.4 NOT MC 3.4	O N/A	9.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Annex B

(normative)

Formal definition of data types using ITU-T Rec. X.208

This annex provides the ASN.1 modules defined for the purpose of this Standard.

B.1 ROSE APDU types

Table B-1 shows the formal definition of the ROSE APDU data types used in the functional protocol.

Table B-1 - ROSE APDU types

```
Remote-Operations-Apdus
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) remote-operations-apdus(1)
}
DEFINITIONS ::=
BEGIN
EXPORTS         InvokeIDType,APDU;
IMPORTS         OPERATION, ERROR
        FROM Remote-Operation-Notation
        {       joint-iso-ccitt remote-operations(4) notation(0)        };


APDU            ::= CHOICE {
                        invokeAPDU              [1]     IMPLICIT        InvokeAPDU,
                        returnResultAPDU        [2]     IMPLICIT        ReturnResultAPDU,
                        returnErrorAPDU         [3]     IMPLICIT        ReturnErrorAPDU,
                        rejectAPDU              [4]     IMPLICIT        RejectAPDU}


InvokeAPDU      ::= SEQUENCE {
                        invokeID                InvokeIDType,
                        linked-ID               [0]     IMPLICIT        InvokeIDType OPTIONAL,
                        operation-value         OPERATION,
                        argument                ANY DEFINED BY          operation-value OPTIONAL}

        -- ANY is filled by the single ASN.1 data type following the keyword
        -- ARGUMENT in the type definition of a particular operation.


InvokeIDType    ::= INTEGER (-32768..32767)

ReturnResultAPDU ::= SEQUENCE {
                        invokeID                InvokeIDType,
                        SEQUENCE {
                                operation-value OPERATION,
                                result          ANY DEFINED BY operation-value } OPTIONAL
                        }


        -- ANY is filled by the single ASN.1 data type following the keyword
        -- RESULT in the type definition of a particular operation.

ReturnErrorAPDU ::= SEQUENCE {
                        invokeID        InvokeIDType,
                        error-value     ERROR,
                        parameter       ANY DEFINED BY
                                        error-value     OPTIONAL}
```

```
        -- ANY is filled by the single ASN.1 data type following the keyword
        -- PARAMETER in the type definition of a particular error


RejectAPDU      ::= SEQUENCE {
                        invokeID            CHOICE {
                                InvokeIDType,
                                NULL},
                        problem CHOICE {
                                [0] IMPLICIT GeneralProblem,
                                [1] IMPLICIT InvokeProblem,
                                [2] IMPLICIT ReturnResultProblem,
                                [3] IMPLICIT ReturnErrorProblem}}

GeneralProblem  ::= INTEGER {                   -- ROSE-provider detected
                        unrecognizedAPDU (0),
                        mistypedAPDU (1),
                        badlyStructuredAPDU (2)}

InvokeProblem   ::= INTEGER {                   -- ROSE-user detected supplementary service entity
                        duplicateInvocation (0),
                        unrecognizedOperation (1),
                        mistypedArgument (2),
                        resourceLimitation (3),
                        initiatorReleasing (4),
                        unrecognizedLinkedID (5),
                        linkedResponseUnexpected (6),
                        unexpectedChildOperation (7)}

ReturnResultProblem     ::= INTEGER {           -- ROSE-user detected
                        unrecognizedInvocation (0),
                        resultResponseUnexpected (1),
                        mistypedResult (2)}

ReturnErrorProblem      ::= INTEGER {           -- ROSE-user detected
                        unrecognizedInvocation (0),
                        errorResponseUnexpected (1),
                        unrecognizedError (2),
                        unexpectedError (3),
                        mistypedParameter (4)}

END             -- of Remote-Operations-Apdus
```

B.2 Definition of embedded B-QSIG information elements

Table B-2 contains the ASN.1 definition of a general applicable type used to include B-QSIG information elements in ASN.1 definitions.

The B-QSIG information elements to be used shall be indicated as comment at the point where the type BqsigInformationElement is used.

Table B-2 - Definition of embedded B-QSIG information elements

```
Bqsig-generic-parameter-definition
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        bqsig-generic-parameters(2) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN
EXPORTS         BqsigInformationElement;

BqsigInformationElement  ::= [APPLICATION 0]        IMPLICIT        OCTET STRING

END     -- of Bqsig-generic-parameter-definition
```

B.3 Network facility extension

Table B-3 contains the ASN.1 definition of type NetworkFacilityExtension.

Table B-3 - Network Facility Extension Coding

```
Network-Facility-Extension
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        network-facility-extension( 3)  }

DEFINITIONS ::=
BEGIN
EXPORTS         NetworkFacilityExtension;
IMPORTS         PartyNumber FROM Addressing-Data-Elements
                {       iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0)  bqsig-generic-procedures (254)
                                                                addressing-data-elements (15)  };

NetworkFacilityExtension        ::=     [10]    IMPLICIT SEQUENCE
                {       sourceEntity                    [0]     IMPLICIT EntityType,
                        sourceEntityAddress             [1]     AddressInformation      OPTIONAL,
                        destinationEntity               [2]     IMPLICIT EntityType,
                        destinationEntityAddress        [3]     AddressInformation      OPTIONAL
                }

EntityType                      ::=     ENUMERATED
                                {       endPINX ( 0),
                                        anyTypeOfPINX ( 1),
                                        reserved1 (2),
                                        reserved2 (3),
                                        reserved3 (4),
                                        reserved4 (5),
                                        reserved5 (6),
                                        reserved6 (7),
                                        reserved7 (8),
                                        reserved8 (9),
                                        reserved9 (10),
                                        reserved10 (11)
                                }

AddressInformation              ::=     PartyNumber

END     -- of Network-Facility-Extension
```

B.4 NOTIFICATION macro and notification for conveying embedded B-QSIG information elements

Table B-4 defines the ASN.1 NOTIFICATION macro used for defining notifications that can be carried in the Notification indicator as defined in 9.2.2.3. It also defines the notification value bqsigIeNotification, the use of which is described in 9.2.2.3.

Table B-4 - Notification macro definition

```
Notification-macro
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254) notification-macro (4)  }

DEFINITIONS ::=
BEGIN
EXPORTS         NOTIFICATION, bqsigIeNotification;
IMPORTS         BqsigInformationElement   FROM    Bqsig-generic-parameter-definition
                {       iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0)  bqsig-generic-procedures (254)
bqsig-generic-parameters (2)   };

NOTIFICATION MACRO   ::=
BEGIN
TYPE NOTATION                   ::= Argument
VALUE NOTATION                  ::= value       ( VALUE CHOICE
                                                {       localValue      INTEGER,
                                                        globalValue     OBJECT IDENTIFIER
                                                }
Argument                        ::=     "ARGUMENT" NamedType
NamedType                       ::=     identifier type | type

END     -- of NOTIFICATION macro


--  this notification is used to convey information elements used as notifications across a PISN

bqsigIeNotification             NOTIFICATION
                                ARGUMENT        BqsigInformationElement
                                ::= 2501

END     -- of Notification-macro
```

B.5 Addressing information definition

Table B-5 contains the definition of ASN.1 types for encoding B-PISN addressing information.

Table B-5 - Addressing data elements

```
Addressing-Data-Elements
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        addressing-data-elements (15)  }

DEFINITIONS ::=
BEGIN
EXPORTS         PresentedAddressScreened,
                PresentedAddressUnscreened,
                PresentedNumberScreened,
                PresentedNumberUnscreened,
                Address,
                PartyNumber,
                PartySubaddress,
                ScreeningIndicator,
                PresentationAllowedIndicator;
```

```
PresentedAddressScreened ::= CHOICE{
        presentationAllowedAddress                  [0]  IMPLICIT AddressScreened,
        presentationRestricted                      [1]  IMPLICIT NULL,
        numberNotAvailableDueToInterworking         [2]  IMPLICIT NULL,
        presentationRestrictedAddress               [3]  IMPLICIT AddressScreened  }

PresentedAddressUnscreened ::= CHOICE{
        presentationAllowedAddress                  [0]  IMPLICIT Address,
        presentationRestricted                      [1]  IMPLICIT NULL,
        numberNotAvailableDueToInterworking         [2]  IMPLICIT NULL,
        presentationRestrictedAddress               [3]  IMPLICIT Address  }

PresentedNumberScreened ::= CHOICE{
        presentationAllowedAddress                  [0]  IMPLICIT NumberScreened,
        presentationRestricted                      [1]  IMPLICIT NULL,
        numberNotAvailableDueToInterworking         [2]  IMPLICIT NULL,
        presentationRestrictedAddress               [3]  IMPLICIT NumberScreened  }

PresentedNumberUnscreened ::= CHOICE{
        presentationAllowedAddress                  [0]  PartyNumber,
        presentationRestricted                      [1]  IMPLICIT NULL,
        numberNotAvailableDueToInterworking         [2]  IMPLICIT NULL,
        presentationRestrictedAddress               [3]  PartyNumber  }

AddressScreened ::= SEQUENCE {
        partyNumber             PartyNumber,
        screeningIndicator      ScreeningIndicator,
        partySubaddress         PartySubaddress OPTIONAL  }

NumberScreened ::= SEQUENCE {
        partyNumber             PartyNumber,
        screeningIndicator      ScreeningIndicator  }

Address ::= SEQUENCE {
        partyNumber             PartyNumber,
        partySubaddress         PartySubaddress OPTIONAL  }

PartyNumber ::= CHOICE{
        unknownPartyNumber      [0]  IMPLICIT  NumberDigits,
                -- the numbering plan is the default numbering plan of the network.
                -- it is recommended that this values is used.
        publicPartyNumber       [1]  IMPLICIT PublicPartyNumber,
                -- the numbering plan is according to Rec. E.163 and
                -- E.164
        nsapEncodedNumber       [2]  IMPLICIT NsapEncodedNumber,
                -- ATM endsystem address encoded as an NSAP address
        dataPartyNumber         [3]  IMPLICIT  NumberDigits,
                -- not used, value reserved
        telexPartyNumber        [4]  IMPLICIT NumberDigits,
                -- not used, value reserved
        privatePartyNumber      [5]  IMPLICIT PrivatePartyNumber,
        nationalStandardPartyNumber  [8]  IMPLICIT NumberDigits  }
                -- not used, values reserved

PublicPartyNumber ::= SEQUENCE {
        publicTypeOfNumber      PublicTypeOfNumber,
        publicNumberDigits      NumberDigits  }

PrivatePartyNumber ::= SEQUENCE {
        privateTypeOfNumber     PrivateTypeOfNumber,
        privateNumberDigits     NumberDigits  }
```

```
NumberDigits            ::=     NumericString   (SIZE (1..20))

PublicTypeOfNumber      ::=     ENUMERATED  {
                        unknown (0),
                                -- if used number digits carry prefix indicating type of
                                -- number according to national recommendations.
                        internationalNumber (1),
                        nationalNumber (2),
                        networkSpecificNumber (3),
                                -- not used, value reserved
                        subscriberNumber (4),
                        abbreviatedNumber (6)   }
                                -- valid only for called party number at the outgoing access,
                                -- network substitutes appropriate number.

PrivateTypeOfNumber     ::=     ENUMERATED  {
                        unknown (0),
                        level2RegionalNumber (1),
                        level1RegionalNumber (2),
                        pISNSpecificNumber (3),
                        localNumber (4),
                        abbreviatedNumber (6)           }

NsapEncodedNumber       ::=     OCTET STRING (SIZE(20))

PartySubaddress         ::=     CHOICE{
                        userSpecifiedSubaddress  UserSpecifiedSubaddress,
                                -- not recommended
                        nSAPSubaddress          NSAPSubaddress                  }
                                -- according to Rec. X.213.

UserSpecifiedSubaddress ::=     SEQUENCE    {
                        subaddressInformation           SubaddressInformation,
                        oddCountIndicator               BOOLEAN     OPTIONAL    }
                                -- used when the coding of subaddress is BCD
                        NSAPSubaddress          ::=     OCTET STRING  (SIZE(1..20))
                                -- specified according to X.213. some networks may limit
                                -- the subaddress value to some other length, e.g. 4 octets.
                        SubaddressInformation   ::=     OCTET STRING  (SIZE(1..20))
                                -- coded according to user requirements. some networks
                                -- may limit the subaddress value to some other length,
                                -- e.g. 4 octets.

ScreeningIndicator      ::=     ENUMERATED  {
                        userProvidedNotScreened (0),
                                -- number was provided by a remote user terminal
                                -- equipment, and has been screened by a network that
                                -- is not the local public or the local private network.
                        userProvidedVerifiedAndPassed (1),
                                -- number was provided by a remote user terminal
                                -- equipment (or by a remote private network), and has
                                -- been screened by the local public or the local private
                                -- network.
                        userProvidedVerifiedAndFailed (2),
                                -- not used, value reserved.
                        networkProvided (3)             }
                                -- number was provided by local public or local private
                                -- network.
```

```
                PresentationAllowedIndicator        ::=         BOOLEAN

END     -- of Addressing-Data-Elements
```

B.6 Interpretation APDU

Table B-6 contains the ASN.1 definition of type Interpretation APDU.

Table B-6 - Interpretation APDU

```
Interpretation-Apdu
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) interpretation-apdu (6)  }

DEFINITIONS ::=
BEGIN
EXPORTS                 InterpretationApdu;

InterpretationApdu          ::=         [11] IMPLICIT ENUMERATED
        {           discardAnyUnrecognisedInvokePdu ( 0),
                    clearCallIfAnyInvokePduNotRecognised ( 1),
                            -- This value also applies to CO-BI connections,
                    rejectAnyUnrecognisedInvokePdu ( 2)
                            -- This coding is implied by the absence of an interpretation APDU

        }
END             -- of Interpretation-Apdu
```

B.7 Notification Data Structure

Table B-7 contains the ASN.1 definition of type NotificationDataStructure.

Table B-7 - ASN.1 encoded Notification Data Structure

```
Notification-Data-Structure
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) notification-data-structure (7)
}

DEFINITIONS ::=
BEGIN
EXPORTS                 NotificationDataStructure;

IMPORTS         NOTIFICATION   FROM    Notification-Macro
        { iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0) bqsig-generic-procedures (254)
                        notification-macro (4)      };
NotificationDataStructure  ::=      SEQUENCE
        {           notificationTypeID          NOTIFICATION,
                    notificationArgument        ANY DEFINED BY
                                                notificationTypeID
        }

        -- ANY is filled by the single ASN.1 data type following the keyword
        -- ARGUMENT in the type definition of a particular notification.

END             -- of Notification-Data-Structure
```

B.8 EXTENSION macro and Extension data type

Table B-8 contains the ASN.1 definition of type Extension and macro EXTENSION.

Table B-8 - Manufacturer specific extension mechanism

```
Manufacturer-specific-service-extension-definition
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) msi-definition (8)  }

DEFINITIONS     ::=
BEGIN
EXPORTS                 Extension, EXTENSION;

EXTENSION MACRO                         ::=
                TYPE NOTATION           ::= Argument
                VALUE NOTATION          ::= Value (VALUE(OBJECT IDENTIFIER))
                Argument                        ::= “Argument” NamedType
                NamedType               ::= identifier type | type

END     -- of EXTENSION macro

Extension               ::=             SEQUENCE
                                        {       manufacturer            EXTENSION,
                                                ANY DEFINED BY          manufacturer
                                        }

END     -- of Manufacturer-specific-service-extension-definition
```

Annex C

(normative)

Formal definition of data types using ITU-T Rec. X.680

This annex provides the ASN.1 modules defined for the purpose of this Standard.

C.1 APDU types

Table C-1 shows the formal definition of the APDU data types used in the functional protocol.

Table C-1 - APDU types

```
Revised-Remote-Operations-Apdus
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254)
                        revised-remote-operations-apdus(11)  }

DEFINITIONS
IMPLICIT TAGS ::=
BEGIN
-- exports everything
IMPORTS OPERATION, ERROR FROM {joint-iso-ccitt remote-operations(4) informationObjects(5) version1(0)};
ROS {InvokeId:InvokeIdSet, OPERATION:Invokable, OPERATION:Returnable} ::=          CHOICE
                {
                        invoke          [1]     Invoke {{InvokeIdSet}, {Invokable}},
                        returnResult    [2]     ReturnResult {{Returnable}},
                        returnError     [3]     ReturnError {{Errors{{Returnable}}}},
                        reject          [4]     Reject
                }
(CONSTRAINED BY { -- must conform to the above definition -- }
! RejectProblem : general-unrecognisedPDU)
Invoke {InvokeId:InvokeIdSet, OPERATION:Operations} ::=     SEQUENCE
                {
                        invokeId InvokeId (InvokeIdSet)
                                                        (CONSTRAINED BY {-- must be unambiguous --}
                                                        ! RejectProblem : invoke-duplicateInvocation),
                        linkedId  CHOICE
                                                {
                                                present  [0]    IMPLICIT        present <       InvokeId,
                                                absent   [1]    IMPLICIT NULL
                                                }
                                                (CONSTRAINED BY {-- must identify an outstanding operation --}
                                                ! RejectProblem : invoke-unrecognisedLinkedId)
                                                (CONSTRAINED BY {-- which has one or more linked operations--}
                                                ! RejectProblem : invoke-linkedResponseUnexpected)
                                                        OPTIONAL,
                        opcode  OPERATION.&operationCode
                                                ({Operations}
                                                ! RejectProblem : invoke-unrecognisedOperation),
                        argument        OPERATION.&ArgumentType
                                                ({Operations} {@opcode}
                                                ! RejectProblem : invoke-mistypedArgument)
                                                                OPTIONAL
                }
(CONSTRAINED BY { -- must conform to the above definition -- }
! RejectProblem : general-mistypedPDU)
(
        WITH COMPONENTS
        {...,
                linkedId  ABSENT
        }
```

```
        WITH COMPONENTS
        {...,
                linkedId  PRESENT,
                opcode
                (CONSTRAINED BY {-- must be in the &Linked field of the associated operation --}
                ! RejectProblem : invoke-unexpectedLinkedOperation)
        }
)
ReturnResult {OPERATION:Operations}::=   SEQUENCE
                {
                        invokeId InvokeId
                                        (CONSTRAINED BY {-- must be that for an outstanding operation  --}
                                        ! RejectProblem : returnResult-unrecognisedInvocation)
                                        (CONSTRAINED BY {-- which returns a result --}
                                        ! RejectProblem : returnResult-resultResponseUnexpected),
                        result    SEQUENCE
                {
                        opcode  OPERATION.&operationCode
                                        (({Operations})(CONSTRAINED BY {-- identified by invokeId --}
                                        ! RejectProblem : returnResult-unrecognisedInvocation)),
                        result    OPERATION.&ResultType
                                        ({Operations} {@.opcode}
                                        ! RejectProblem : returnResult-mistypedResult)
                }
                                OPTIONAL
                }
(CONSTRAINED BY { -- must conform to the above definition -- }
! RejectProblem : general-mistypedPDU)
ReturnError {ERROR:Errors} ::=      SEQUENCE
                {
                        invokeId InvokeId
                                        (CONSTRAINED BY {-- must be that for an outstanding operation --}
                                        ! RejectProblem : returnError-unrecognisedInvocation)
                                        (CONSTRAINED BY {-- which returns an error --}
                                        ! RejectProblem : returnError-errorResponseUnexpected),
                        errcode  ERROR.&errorCode
                                                ({Errors}
                                                ! RejectProblem : returnError-unrecognisedError)
                                                (CONSTRAINED BY
                                                        {--must be in the &Errors field of the associated opÕn --}
                                                        ! RejectProblem : returnError-unexpectedError),
                        parameter ERROR.&ParameterType
                                                ({Errors}{@errcode}
                                                ! RejectProblem : returnError-mistypedParameter) OPTIONAL
                }
(CONSTRAINED BY { -- must conform to the above definition -- }
! RejectProblem : general-mistypedPDU)
Reject ::=          SEQUENCE
                {
                        invokeId InvokeId,
                        problem CHOICE
                                        {
                                        general             [0]     GeneralProblem,
                                        invoke              [1]     InvokeProblem,
                                        returnResult        [2]     ReturnResultProblem,
                                        returnError         [3]     ReturnErrorProblem
                                        }
                }
```

```
(CONSTRAINED BY { -- must conform to the above definition -- }
! RejectProblem : general-mistypedPDU)
GeneralProblem          ::=     INTEGER
                {
                        unrecognisedComponent (0),
                        mistypedComponent (1),
                        badlyStructuredComponent (2)
                }
InvokeProblem           ::=     INTEGER
                {
                        duplicateInvocation (0),
                        unrecognisedOperation (1),
                        mistypedArgument (2),
                        resourceLimitation (3),
                        releaseInProgress (4),
                        unrecognisedLinkedId (5),
                        linkedResponseUnexpected (6),
                        unexpectedLinkedOperation (7),
                }
ReturnResultProblem     ::=     INTEGER
                {
                        unrecognisedInvocation (0),
                        resultResponseUnexpected (1),
                        mistypedResult (2)
                }
ReturnErrorProblem      ::=     INTEGER
                {
                        unrecognisedInvocation (0),
                        errorResponseUnexpected (1),
                        unrecognisedError (2),
                        unexpectedError (3),
                        mistypedParameter (4)
                }
RejectProblem           ::=     INTEGER
                {
                        general-unrecognisedPDU (0),
                        general-mistypedPDU (1),
                        general-badlyStructuredPDU (2),
                        invoke-duplicateInvocation (10),
                        invoke-unrecognisedOperation (11),
                        invoke-mistypedArgument (12),
                        invoke-resourceLimitation (13),
                        invoke-releaseInProgress (14),
                        invoke-unrecognisedLinkedId (15),
                        invoke-linkedResponseUnexpected (16),
                        invoke-unexpectedLinkedOperation (17),
                        returnResult-unrecognisedInvocation (20),
                        returnResult-resultResponseUnexpected (21),
                        returnResult-mistypedResult (22),
                        returnError-unrecognisedInvocation (30),
                        returnError-errorResponseUnexpected (31),
                        returnError-unrecognisedError (32),
                        returnError-unexpectedError (33),
                        returnError-mistypedParameter (34)
                }
InvokeId                ::=     CHOICE
                {
                        present  INTEGER,
                        absent   NULL
                }
```

```
noInvokeId InvokeId        ::=     absent:NULL
NoInvokeId InvokeId        ::=     {noInvokeId}
Errors {OPERATION:Operations} ERROR ::=          {Operations.&Errors}
END -- end of generic ROS PDU definitions
```

C.2 Definition of embedded B-QSIG information elements

Table C-2 contains the ASN.1 definition of a general applicable type used to include B-QSIG information elements in ASN.1 definitions.

The B-QSIG information elements to be used shall be indicated as comment at the point where the type BqsigInformationElement is used.

Table C-2 - Definition of embedded B-QSIG information elements

```
Bqsig-generic-parameter-definition
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        bqsig-generic-parameters(12) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN
EXPORTS         BqsigInformationElement;
BqsigInformationElement  ::= [APPLICATION 0] IMPLICIT OCTET STRING
END -- of Bqsig-generic-parameter-definition
```

C.3 Network facility extension

Table C-3 contains the ASN.1 definition of type NetworkFacilityExtension.

Table C-3 - Network Facility Extension Coding

```
Network-Facility-Extension
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        network-facility-extension(13)  }

DEFINITIONS ::=
BEGIN
EXPORTS         NetworkFacilityExtension;
IMPORTS         PartyNumber FROM Addressing-Data-Elements
                {       iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0)  bqsig-generic-procedures (254)
                                                        addressing-data-elements (15 )  };

NetworkFacilityExtension        ::=     [10]    IMPLICIT SEQUENCE
                {       sourceEntity                    [0]     IMPLICIT EntityType,
                        sourceEntityAddress             [1]     AddressInformation      OPTIONAL,
                        destinationEntity               [2]     IMPLICIT EntityType,
                        destinationEntityAddress        [3]     AddressInformation      OPTIONAL
                }

EntityType                      ::=     ENUMERATED
                                {       endPINX ( 0),
                                        anyTypeOfPINX ( 1)
                                        reserved1 (2),
                                        reserved2 (3),
                                        reserved3 (4),
                                        reserved4 (5),
                                        reserved5 (6),
                                        reserved6 (7),
                                        reserved7 (8),
                                        reserved8 (9),
                                        reserved9 (10),
```

```
                                        reserved10 (11)
                                }

AddressInformation              ::=     PartyNumber

END     -- of Network-Facility-Extension
```

C.4 NOTIFICATION object class and notification for conveying embedded B-QSIG information elements

Table C-4 defines the ASN.1 NOTIFICATION object class used for defining notifications that can be carried in the Notification indicator as defined in 9.2.2.3. It also defines the notification bqsigIeNotification, the use of which is described in 9.2.2.3.

Table C-4 - Notification object class definition

```
Notification-object-class
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                        notification-object-class (14)  }

DEFINITIONS ::=
BEGIN
EXPORTS         NOTIFICATION, bqsigIeNotification;
IMPORTS         BqsigInformationElement   FROM    Bqsig-generic-parameter-definition
                {       iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0)  bqsig-generic-procedures (254)
                                        bqsig-generic-parameters (12)   };

NOTIFICATION ::= CLASS
{
        &ArgumentType           OPTIONAL,
        &argumentTypeOptional   BOOLEAN OPTIONAL,
        &notificationCode       Code UNIQUE
}
WITH SYNTAX
{
        [ARGUMENT               &ArgumentType           [OPTIONAL       &argumentTypeOptional]]
        CODE                    &notificationCode
}

Code ::= CHOICE
{
        local   INTEGER,
        global  OBJECT IDENTIFIER
}


--  the notification below is used to convey information elements used as notifications across a PISN

bqsigIeNotification         NOTIFICATION ::=
{
        ARGUMENT        BqsigInformationElement
        CODE            local: 2501
}
END     -- of Notification-object-class
```

C.5 Addressing information definition

Table C-5 contains the definition of ASN.1 types for encoding B-PISN addressing information

Table C-5 - Addressing information definitions

```
Addressing-Data-Elements
        {iso(1) identified-organisation(3) icd-ecma(0012) standard (0) bqsig-generic-procedures (254)
                addressing-data-elements (15)  }

DEFINITIONS ::=
BEGIN
EXPORTS         PresentedAddressScreened,
                PresentedAddressUnscreened,
                PresentedNumberScreened,
                PresentedNumberUnscreened,
                Address,
                PartyNumber,
                PartySubaddress,
                ScreeningIndicator,
                PresentationAllowedIndicator;

                PresentedAddressScreened        ::=     CHOICE{
                        presentationAllowedAddress              [0]     IMPLICIT AddressScreened,
                        presentationRestricted                  [1]     IMPLICIT NULL,
                        numberNotAvailableDueToInterworking     [2]     IMPLICIT NULL,
                        presentationRestrictedAddress           [3]     IMPLICIT AddressScreened        }

                PresentedAddressUnscreened      ::=     CHOICE{
                        presentationAllowedAddress              [0]     IMPLICIT Address,
                        presentationRestricted                  [1]     IMPLICIT NULL,
                        numberNotAvailableDueToInterworking     [2]     IMPLICIT NULL,
                        presentationRestrictedAddress           [3]     IMPLICIT Address        }

                PresentedNumberScreened         ::=     CHOICE{
                        presentationAllowedAddress              [0]     IMPLICIT NumberScreened,
                        presentationRestricted                  [1]     IMPLICIT NULL,
                        numberNotAvailableDueToInterworking     [2]     IMPLICIT NULL,
                        presentationRestrictedAddress           [3]     IMPLICIT NumberScreened         }

                PresentedNumberUnscreened       ::=     CHOICE{
                        presentationAllowedAddress              [0]     PartyNumber,
                        presentationRestricted                  [1]     IMPLICIT NULL,
                        numberNotAvailableDueToInterworking     [2]     IMPLICIT NULL,
                        presentationRestrictedAddress           [3]     PartyNumber     }

                AddressScreened         ::=     SEQUENCE        {
                                        partyNumber             PartyNumber,
                                        screeningIndicator      ScreeningIndicator,
                                        partySubaddress PartySubaddress OPTIONAL        }

                NumberScreened          ::=     SEQUENCE        {
                                        partyNumber             PartyNumber,
                                        screeningIndicator      ScreeningIndicator      }

                Address                 ::=     SEQUENCE        {
                                        partyNumber             PartyNumber,
                                        partySubaddress PartySubaddress OPTIONAL        }

                PartyNumber             ::=     CHOICE{
                                        unknownPartyNumber      [0]     IMPLICIT  NumberDigits,
                                                -- the numbering plan is the default numbering plan of the network.
```

```
                -- it is recommended that this values is used.
        publicPartyNumber       [1]     IMPLICIT PublicPartyNumber,
                -- the numbering plan is according to Rec. E.163 and
                -- E.164
        nsapEncodedNumber       [2]     IMPLICIT NsapEncodedNumber,
                -- ATM endsystem address encoded as an NSAP address
        dataPartyNumber [3]     IMPLICIT        NumberDigits,
                -- not used, value reserved
        telexPartyNumber        [4]     IMPLICIT NumberDigits,
                -- not used, value reserved
        privatePartyNumber      [5]     IMPLICIT PrivatePartyNumber,
        nationalStandardPartyNumber     [8]     IMPLICIT NumberDigits   }
                -- not used, values reserved

PublicPartyNumber       ::=     SEQUENCE        {
        publicTypeOfNumber      PublicTypeOfNumber,
        publicNumberDigits      NumberDigits    }

PrivatePartyNumber      ::=     SEQUENCE        {
        privateTypeOfNumber     PrivateTypeOfNumber,
        privateNumberDigits     NumberDigits    }

NumberDigits            ::=     NumericString   (SIZE (1..20))

PublicTypeOfNumber      ::=     ENUMERATED      {
        unknown (0),
                -- if used number digits carry prefix indicating type of
                -- number according to national recommendations.
        internationalNumber (1),
        nationalNumber (2),
        networkSpecificNumber (3),
                -- not used, value reserved
        subscriberNumber (4),
        abbreviatedNumber (6)   }
                -- valid only for called party number at the outgoing access,
                -- network substitutes appropriate number.

PrivateTypeOfNumber     ::=     ENUMERATED      {
        unknown (0),
        level2RegionalNumber (1),
        level1RegionalNumber (2),
        pISNSpecificNumber (3),
        localNumber (4),
        abbreviatedNumber (6)           }

NsapEncodedNumber       ::=     OCTET STRING (SIZE(20))

PartySubaddress         ::=     CHOICE{
        userSpecifiedSubaddress  UserSpecifiedSubaddress,
                -- not recommended
        nSAPSubaddress          NSAPSubaddress                  }
                -- according to Rec. X.213.

UserSpecifiedSubaddress ::=     SEQUENCE        {
        subaddressInformation           SubaddressInformation,
        oddCountIndicator               BOOLEAN         OPTIONAL        }
                -- used when the coding of subaddress is BCD
        NSAPSubaddress          ::=     OCTET STRING  (SIZE(1..20))
                -- specified according to X.213. some networks may limit
                -- the subaddress value to some other length, e.g. 4 octets.
        SubaddressInformation   ::=     OCTET STRING  (SIZE(1..20))
```

```
                                        -- coded according to user requirements. some networks
                                        -- may limit the subaddress value to some other length,
                                        -- e.g. 4 octets.

        ScreeningIndicator          ::=     ENUMERATED   {
                                    userProvidedNotScreened (0),
                                        -- number was provided by a remote user terminal
                                        -- equipment, and has been screened by a network that
                                        -- is not the local public or the local private network.
                                    userProvidedVerifiedAndPassed (1),
                                        -- number was provided by a remote user terminal
                                        -- equipment (or by a remote private network), and has
                                        -- been screened by the local public or the local private
                                        -- network.
                                    userProvidedVerifiedAndFailed (2),
                                        -- not used, value reserved.
                                    networkProvided (3)                 }
                                        -- number was provided by local public or local private
                                        -- network.

        PresentationAllowedIndicator        ::=     BOOLEAN

END     -- of Addressing-Data-Elements
```

C.6 Interpretation APDU

Table C-6 contains the ASN.1 definition of type Interpretation APDU

Table C-6 - Interpretation APDU

```
Interpretation-Apdu
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) interpretation-apdu (16)  }

DEFINITIONS ::=
BEGIN
EXPORTS                 InterpretationApdu;

InterpretationApdu          ::=     [11] IMPLICIT ENUMERATED
        {       discardAnyUnrecognisedInvokePdu ( 0),
                clearCallIfAnyInvokePduNotRecognised ( 1),
                        -- This value also applies to CO-BI connections,
                rejectAnyUnrecognisedInvokePdu ( 2)
                        -- This coding is implied by the absence of an interpretation APDU

        }
END             -- of Interpretation-Apdu
```

C.7 Notification Data Structure

Table C-7 contains the ASN.1 definition of type NotificationDataStructure.

Table C-7 - ASN.1 encoded Notification Data Structure

```
Notification-Data-Structure
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) notification-data-structure
(17)  }

DEFINITIONS ::=
BEGIN
EXPORTS                 NotificationDataStructure{};
```

```
IMPORTS          NOTIFICATION   FROM    Notification-object-class
        {iso ( 1) identified-organisation(3) icd-ecma(0012) standard ( 0) bqsig-generic-procedures (254)
                                notification-object-class (14)};
NotificationDataStructure {NOTIFICATION:NotificationSet}      ::=        SEQUENCE
        {
                notificationValue           NOTIFICATION.&notificationCode
                                                ({NotificationSet}),
                notificationArgument        NOTIFICATION.&ArgumentType
                                                ({NotificationSet}{@notificationValue})
                                                OPTIONAL
        }
-- NotificationSet is a set of objects of class NOTIFICATION. Element notificationValue is constrained
-- to be the identifier of an object from that set, and element notificationArgument is constrained to
-- be the argument type for that particular object.

END             -- of Notification-Data-Structure
```

C.8 EXTENSION macro and Extension data type

Table C-8 contains the ASN.1 definition of the EXTENSION object class and type Extension.

Table C-8 - Manufacturer specific extension mechanism

```
Manufacturer-specific-service-extension-definition
    { iso(1) identified-organisation(3) icd-ecma(0012) standard(0) bqsig-generic-procedures (254) msi-definition (18)  }

DEFINITIONS     ::=
BEGIN
EXPORTS                 Extension, EXTENSION{};

EXTENSION ::= CLASS
{
        &ArgumentType,
        &extensionId                    OBJECT IDENTIFIER
}
WITH SYNTAX
{
        ARGUMENT                &ArgumentType
        IDENTIFIER              &extensionId
}

Extension {EXTENSION:ExtensionSet} ::= SEQUENCE
        {
                extensionId             EXTENSION.&extendionId
                                                        ({Extensionset})
                extensionArgument       EXTENSION.&ArgumentType
                                                        ({ExtensionSet}{@extensionId})
        }
-- ExtensionSet is a set of objects of class EXTENSION. Element extensionId is constrained to be
-- the identifier of an object from that set, and element extensionArgument is constrained to be the
-- argument type for that particular object.

END     -- of Manufacturer-specific-service-extension-definition
```

Annex D

(informative)

Information flows

D.1 Connection-oriented bearer independent transport mechanism

D.1.1 Bearer independent establishment and data transfer

Bearer independent establishment and data transfer described in ITU-T Rec. Q.2932.1 Appendix I clause I.1.1 shall apply.

Annex E

(informative)

Instruction indicators

Instruction indicators described in ITU-T Rec. Q.2932.1 Appendix II apply.

Annex F

(informative)

Formal definitions of remote operations notation using ITU-T Rec. X.208

Table F-1 - Formal definition of remote operations data types (extract of ITU-T Rec. X.219 Figure 4/X.219)

```
Remote-Operation-Notation {joint-iso-ccitt remote-operations(4) notation(0)}

DEFINITIONS ::=

BEGIN

EXPORTS      OPERATION, ERROR;

-- macro definition for operations
OPERATION MACRO ::=
BEGIN
TYPE NOTATION               ::= Argument Result Errors LinkedOperations
VALUE NOTATION              ::= value (VALUE CHOICE {
                                        localValue INTEGER,
                                        globalValue OBJECT IDENTIFIER}

Argument            ::= "ARGUMENT" NamedType | empty

Result              ::= "RESULT" ResultType | empty

ResultType          ::= NamedType | empty

Errors              ::= "ERRORS" "{" ErrorNames "}" | empty

LinkedOperations    ::= "LINKED" "{" LinkedOperationNames "}" | empty

ErrorNames          ::= ErrorList | empty

ErrorList           ::= Error | ErrorList "," Error

Error               ::= value (ERROR)
                            | type                  -- shall reference an error type if no error
                            -- value is specified


LinkedOperationNames  ::= OperationList | empty

OperationList         ::= Operation | OperationList "," Operation

Operation             ::= value (OPERATION)
                            | type                  -- shall reference an operation type if no
                                                    -- operation value is specified
```

```
NamedType               ::= identifier type | type

END -- of OPERATION MACRO

-- macro definition for operations errors

ERROR MACRO ::=

BEGIN

TYPE NOTATION           ::= Parameter

VALUE NOTATION          ::= value (VALUE CHOICE {
                                                    localValue INTEGER,
                                                    globalValue OBJECT IDENTIFIER})

Parameter               ::= "PARAMETER" NamedType | empty

NamedType               ::= identifier type | type

END -- of ERROR MACRO

END -- end of Remote-Operation-Notation
```

Annex G

(informative)

Formal definitions of remote operations notation using ITU-T Rec. X.680

Table G-1 - Formal definition of remote operations data types (extract of ITU-T Rec. X.880 annex A)

```
Remote-Operations-Information-Objects
        {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)}

DEFINITIONS ::=
BEGIN
-- exports everything
IMPORTS emptyBind, emptyUnbind
                FROM {joint-iso-ccitt remote-operations(4) useful-definitions(7) version1(0)}

OPERATION ::=   CLASS
                {
                    &ArgumentType                       OPTIONAL,
                    &argumentTypeOptional   BOOLEAN     OPTIONAL,
                    &returnResult           BOOLEAN     DEFAULT TRUE,
                    &ResultType                         OPTIONAL,
                    &resultTypeOptional     BOOLEAN     OPTIONAL,
                    &Errors                 ERROR       OPTIONAL,
                    &Linked                 OPERATION   OPTIONAL,
                    &synchronous            BOOLEAN     DEFAULT FALSE,
                    &alwaysReturns          BOOLEAN     DEFAULT TRUE,
                    &InvokePriority         Priority    OPTIONAL,
                    &ResultPriority         Priority    OPTIONAL,
                    &operationCode          Code UNIQUE OPTIONAL
                }
WITH SYNTAX
                {
                    [ARGUMENT               &ArgumentType   [OPTIONAL   &argumentTypeOptional]]
                    [RESULT                 &ResultType     [OPTIONAL   &resultTypeOptional]]
                    [RETURN RESULT          &returnResult]
                    [ERRORS                 &Errors]
                    [LINKED                 &Linked]
                    [SYNCHRONOUS            &synchronous]
                    [ALWAYS RESPONDS        &alwaysReturns]
                    [INVOKE PRIORITY        &InvokePriority]
                    [RESULT PRIORITY        &ResultPriority]
                    [CODE                   &operationCode]
                }
```

```
ERROR ::=           CLASS
                    {
                        &ParameterType                      OPTIONAL,
                        &parameterTypeOptional  BOOLEAN     OPTIONAL,
                        &ErrorPriority          Priority    OPTIONAL,
                        &errorCode              Code UNIQUE OPTIONAL
                    }
WITH SYNTAX
                    {
                        [PARAMETER              &ParameterType [OPTIONAL      &parameterTypeOptional]]
                        [PRIORITY               &ErrorPriority]
                        [CODE                   &errorCode]
                    }

OPERATION-PACKAGE ::=       CLASS
                    {
                        &Both                   OPERATION   OPTIONAL,
                        &Consumer               OPERATION   OPTIONAL,
                        &Supplier               OPERATION   OPTIONAL,
                        &id                     OBJECT IDENTIFIER UNIQUE OPTIONAL
                    }
WITH SYNTAX
                    {
                        [OPERATIONS             &Both]
                        [CONSUMER INVOKES       &Supplier]
                        [SUPPLIER INVOKES       &Consumer]
                        [ID                     &id]
                    }

CONNECTION-PACKAGE ::=      CLASS
                    {
                        &bind                   OPERATION   DEFAULT emptyBind,
                        &unbind                 OPERATION   DEFAULT emptyUnbind,
                        &responderCanUnbind     BOOLEAN     DEFAULT FALSE,
                        &unbindCanFail          BOOLEAN     DEFAULT FALSE,
                        &id                     OBJECT IDENTIFIER UNIQUE OPTIONAL
                    }
WITH SYNTAX
                    {
                        [BIND                   &bind]
                        [UNBIND                 &unbind]
                        [RESPONDER UNBIND       &responderCanUnbind]
                        [FAILURE TO UNBIND      &unbindCanFail]
                        [ID                     &id]
                    }

CONTRACT ::=        CLASS
                    {
                        &connection             CONNECTION-PACKAGE OPTIONAL,
                        &OperationsOf           OPERATION-PACKAGE OPTIONAL,
                        &InitiatorConsumerOf    OPERATION-PACKAGE OPTIONAL,
                        &InitiatorSupplierOf    OPERATION-PACKAGE OPTIONAL,
                        &id                     OBJECT IDENTIFIER UNIQUE OPTIONAL
                    }
WITH SYNTAX
                    {
                        [CONNECTION                 &connection]
                        [OPERATIONS OF              &OperationsOf]
                        [INITIATOR CONSUMER OF      &InitiatorConsumerOf]
                        [RESPONDER CONSUMER OF      &InitiatorSupplierOf]
                        [ID                         &id]
                    }
```

```
ROS-OBJECT-CLASS ::=  CLASS
                      {
                           &Is                        ROS-OBJECT-CLASS OPTIONAL,
                           &Initiates                 CONTRACT OPTIONAL,
                           &Responds                        CONTRACT OPTIONAL,
                           &InitiatesAndResponds      CONTRACT OPTIONAL,
                           &id                        OBJECT IDENTIFIER UNIQUE
                      }
WITH SYNTAX
                      {
                           [IS                        &Is]
                           [BOTH                      &InitiatesAndResponds]
                           [INITIATES                 &Initiates]
                           [RESPONDS                  &Responds]
                           ID                         &id
                      }

Code ::=         CHOICE
                      {
                           local                      INTEGER,
                           global                     OBJECT IDENTIFIER
                      }

Priority ::=     INTEGER (0..MAX)

END -- end of Information Object specifications
```

Annex H

(informative)

Examples of the use of Manufacturer Specific Information

H.1 Manufacturer Specific Object Identifier in Operation Values

As defined in 14.1, manufacturer who wish to provide manufacturer specific Supplementary services in a standardised manner should use unique operation values, constructed using manufacturer specific object identifier.

Manufacturer specific object identifiers may be constructed in the following way. Manufacturer requiring an assigned identification may apply to a "Sponsoring and Issuing organisation" according to ISO/IEC 6523 and be assigned an organisation identifier. The manufacturer should then use that organisation identifier in an object identifier (as the root of the manufacturer specific service operation value) according to the structure defined by the issuing organisation.

One example of regional Sponsoring and issuing organisation is ECMA, which has been assigned an International Code Designator (ICD). ECMA will assign values to ECMA member companies in its object identifier root. The use of ECMA issued organisation identifiers in object identifiers is as shown in table H-1. PINXs conforming to this International Standard can make use of an organisation identifier issued by any "sponsoring and issuing organisation" (e.g. ECMA or a national body).

Thus, according to table H-1, the ECMA object identifier for a company with the assigned organisation code '1999' (all organisation codes issued by ECMA have 4 digits of which the first is always '1'), may be structured as shown in table H-2. The contents of level 6 is manufacturer specific and may identify a company specific operation value or may not exist at all. In this example, level 6 provides a manufacturer specific operation value.

This object identifier value would then be used in the definitions of the manufacturer specific operation (internally to that manufacturer) An example of a manufacturer specific operation is shown in table H-3.

Table H-1 - Structure of ECMA Object Identifier

level 1:	iso(1)
level 2:	identified-organisation (3)
level 3:	icd-ecma (0012)
level 4:	a) standard (0)
	b) technical-report (1)
	c) member-company (2)
	d) private-ISDN-signalling-domain (9)
level 5:	for c) of level 4: organisation identifier assigned by ECMA
level 6:	this level and others below it are used to suit the purpose of the organisation assigned the value at level 5.

Table H-2 - ECMA Object identifier for hypothetical manufacturer specific service operation

```
Object identifier for hypothetical manufacturer specific
service operation value:

Hypothetical ManufacturerSpecificsupplementaryService ::=
      { iso( 1) identifier-organisation( 3) icd-ecma( 0012)
            member-company( 2) hypothetical-manufacturer( 1999)
                  hypothetical-manufacturer-service( 1)  }

In pure numeric value, this would be:

{ 1 3 0012 2 1999 }

(This shall be encoded as described in ITU-T Rec. X.209)
```

Table H-3 - Example of manufacturer specific operation

```
Hypothetical-service-operation
{     iso   identified-organisation icd-ecma-member-company
            hypothetical-manufacturer hypothetical-service-offering       }

DEFINITIONS ::=
BEGIN

IMPORT  OPERATION FROM Remote-Operation-Notation
      {     joint-iso-ccitt ( 2) remote-operations ( 4) notation ( 0)      };

      hypotheticalService       OPERATION
            ARGUMENT    HypotheticalArgument
            RESULT      HypotheticalResult
            ::=   { iso ( 1) idenitified-organisation ( 3) icd-ecma ( 0012)
                  member -company ( 2) hypothetical-manufacturer ( 1999)
                  hypothetical-manufacturer-service ( 1)        }

      HypotheticalArgument    ::= INTEGER
                  {       hypotheticalParameter1 ( 0),
                          hypotheticalParameter2 ( 1)
                  }

      HypotheticalResult      ::= INTEGER
                  {       hypotheticalResult1 ( 0),
                          hypotheticalResult2 ( 1)
                  }

END         -- of hypothetical-manufacturer-service-operation
```

H.2 Manufacturer specific extensions to standardised Supplementary services

An example of the use the element of type Extension (defined in 14.1) in a standardised Supplementary services definition is given in table H-4 for a hypothetical ISO standard number ‘2222222’. In the operation definitions for standardised supplementary services, the following constructs are used:

– where the standardised parameter (argument of invoke APDU, (result of return result APDU) is a single value (e.g. INTEGER), the Standard can instead specify a SEQUENCE containing a CHOICE of an element of type Extension or a SEQUENCE of elements of type Extension. Thus, the parameter would then become:

```
Parameter     ::=   CHOICE
          {
          INTEGER,
          SEQUENCE
               }
               INTEGER,
               CHOICE
                    {
                    [0]   IMPLICIT Extension,
                    [1]   IMPLICIT SEQUENCE OF
                                   Extension
                    } OPTIONAL
               }
          }
```

– where the parameter is a SEQUENCE type, this would be replaced by a SEQUENCE containing a CHOICE of an element of type Extension or a SEQUENCE of elements of type Extension. Thus, the parameter would then become:

```
Parameter     ::=   SEQUENCE
          {
          List-of-Standard-parameter-types,
          CHOICE
               {
               [0]   IMPLICIT      Extension,
               [1]   IMPLICIT      SEQUENCE OF
                                   Extension
               } OPTIONAL
          }
```

– where there is no defined parameter, a parameter should be added as shown below:

```
Parameter     ::=   CHOICE
          {
          NULL,
          [0]   IMPLICIT           Extension,
          [1]   IMPLICIT           SEQUENCE OF Extension
          }
```

NOTE
The use of implicit tagging within the CHOICE construct containing elements of type Extension should be used consistent with the context specific tags used in the remainder of the SEQUENCE in which it is contained.

In this way, manufacturer specific additions to standardised Supplementary services or additional basic call capability may be included in a generic and backwards compatible manner. the manufacturer object identifier (shown in table H-3 above) should be encoded in the same manner as described in 14.1.

the use of sequence of elements of type Extension allows the coexistence of a number of different extensions to standardised Supplementary service or basic call capabilities. It also allows for future versions of the standardised service to be backwards compatible with, and to coexist with, manufacturer specific additions to the original supplementary service or additional basic call capability.

Table H-4 - Example definition of standardised operation with elements of type extension

```
        Hypothetical-service-operation
        {   iso  standard hypothetical-Standard ( 2222222) first-and-only-module ( 0) }

DEFINITIONS ::=
BEGIN

IMPORT  OPERATION FROM Remote-Operation-Notation
        {   joint-iso-ccitt ( 2) remote-operations ( 4) notation ( 0)    };
        Extension FROM Manufacturer-specific-service-extension-definition
        {   iso standard bqsig-generic-procedures (254) msi-definition ( 0)   };

        hypotheticalService     OPERATION
                ARGUMENT CHOICE
                    {  NormalIntegerArgument,
                       SEQUENCE
                       { NormalIntegerArgument,
                                extension       CHOICE
                                {       [2]     IMPLICIT Extension
                                        [3]     IMPLICIT SEQUENCE OF Extension
                                }       OPTIONAL
                       }
                    }
                RESULT      SEQUENCE
                    {  ListOfNormalResultSequenceElements,
                       extension        CHOICE
                       {        [2]     IMPLICIT Extension
                                [3]     IMPLICIT SEQUENCE OF Extension
                       }        OPTIONAL
                    }
                    ::=         { iso standard hypothetical-standard ( 2222222)
                                        hypothetical-operation ( 10)        }

        NormalIntegerArgument   ::=     INTEGER
                    {  hypotheticalParameter1 ( 0),
                       hypotheticalParameter1 ( 1)
                    }

        ListOfNormalResultSequenceElements      ::=     SEQUENCE    {
                    { normalResultSequenceElement1      [0]     IMPLICIT INTEGER
                    { normalResultSequenceElement2      [1]     IMPLICIT INTEGER    }

END         -- of hypothetical-service-operation
```

Annex I

(informative)

Remote operations protocol

The remote operations (RO) protocol is defined in ITU-T Rec. X.219/X.229 using ASN.1 as specified in ITU-T recommendation X.208 and in X.880 using ASN.1 as specified in ITU-T recommendation X.680. The generic procedures defined in this Standard provide an encoding mechanism for the transport and use of this RO protocol in the B-PISN environment for the provision of Supplementary services or additional basic call capabilities.

In the OSI environment, communication between application processes is represented in terms of communication between a pair of application entities (AEs). Communication between application entities are inherently interactive. Typically, one entity requests that a particular operation be performed; the other entity attempts to perform the operation and then reports the outcome of the attempts. The concept of Remote Operation is a vehicle for supporting interactive applications of this type.

The generic structure of an operation is an elementary request/reply interaction. Operations are carried out within the context of an application-association.

Figure I-1 models the view.

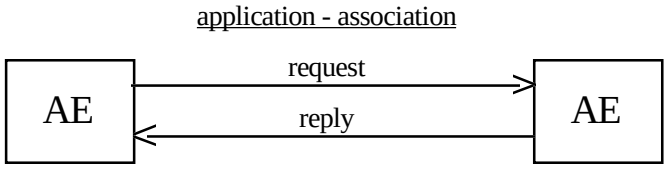


Figure I-1 - Remote operation model

Operations invoked by one AE (the invoker) are performed by the other AE (the performer). Operations may be classified according to whether the performer of an operation is expected to report its outcome:

– in the case of success or failure (a result reply is returned if the operation is successful, an error reply is returned if the operation is unsuccessful);

– in case of failure only (no reply is returned if the operation is successful, an error reply is returned if the operation is unsuccessful);

– in case of success only (a result reply is returned if the operation is successful, no error reply is returned if the operation is unsuccessful);

– or not at all (neither a result nor an error reply is returned, whether the operation was successful or not).

Operations may also be classified according to two possible operation modes: synchronous, in which the invoker requires a reply from the performer before invoking another operation; and asynchronous, in which the invoker may continue to invoke further operations without awaiting a reply.

The following Operation Classes are defined:

Operation Class 1: Synchronous, reporting success or failure (result or error).

Operation Class 2: Asynchronous, reporting success or failure (result or error).

Operation Class 3: Asynchronous, reporting failure (error) only, if any.

Operation Class 4: Asynchronous, reporting success (result) only.

Operation Class 5: Asynchronous, outcome not reported.

The Operation Class of each operation has to be agreed between application entities (e.g. in an Application Protocol Standard).

In some cases, it is useful to group operations into a set of linked operations comprising one parent operation and one or more child operations. The performer of the parent operation may invoke none, one, or more child operations during the execution of

the parent operation. The invoker of the parent operation is the performer of the child operations. A child operation may be a parent operation of another set of linked operations in a recursive manner. Figure I-2 models this concept.

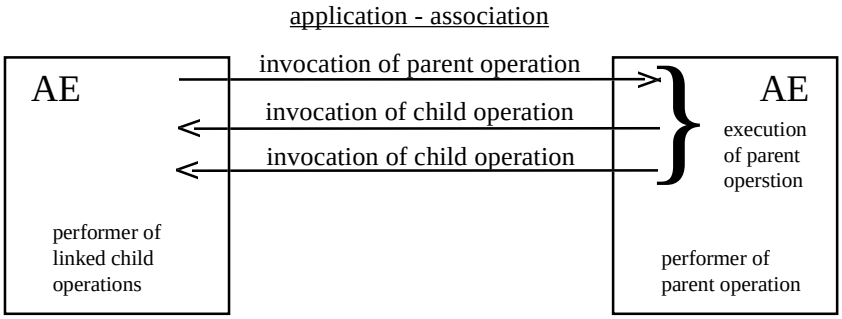


Figure I-2 - Linked operations

An application association defines the relationship between a pair of AEs, and is formed by the exchange of application (in this case Supplementary services) Protocol Control information through the use of the services of underlying layers. The AE that initiates an association is called the association initiating AE, or the association initiator, while the AE that responds to the initiation of an application association by another AE is called the association responding AE, or the association responder.

NOTE
In the application of ROSE for the support of Supplementary services or additional basic call capabilities in B-QSIG the underlying services used by ROSE are those provided by GFT-Control. No use is made of the services of the Reliable Transport Service Element (RTSE). Application associations are classified by which application-entity is allowed to invoke operations:

Association Class 1: Only the association-initiating application-entity can invoke operations.

Association Class 2: Only the association-responding application-entity can invoke operations.

Association Class 3: Only the association-initiating and the association-responding application-entity can invoke operations.

This Standard assumes Application associations of Association Class 3.

Annex J

(informative)

Problem code definitions

Table J-1 - Problem code definitions

General-problem:		
–	unrecognizedAPDU	signifies that the type of the APDU as evidenced by its type identifier, is not one of the four defined in annex B, clause B.1 of this Standard
–	mistypedAPDU	signifies that the structure of the APDU does not conform to annex B, clause B.1 of this Standard
–	badlyStructuredAPDU	signifies that the structure of the APDUs does not conform to the standard notation and encoding, defined in ITU-T Rec. X.208 and X.209, or ITU-T Rec.s X.680 and X.690 as appropriate
Invoke-problem:		
–	duplicateInvocation	signifies that the invoke-identifier parameter violates the assignment rules of ITU-T Rec. X.219.
–	unrecognizedOperation	signifies that the operation is not one of those supported.
–	mistypedArgument	signifies that the type of the operation argument supplied is not expected.
–	resourceLimitation	the performing PINX is not able to perform the invoked operation due to resource limitation.
–	initiatorReleasing	the application is not willing to perform the invoked operation because it is about to attempt to release the connection oriented transport mechanism.
–	unrecognizedLinkedId	signifies that there is no operation in progress with an invoke-identifier equal to the specified linked-identifier.
–	linkedResponseUnexpected	signifies that the invoked operation referred to by linked-identifier is not a parent-operation.
–	unexpectedChildOperation	signifies that the invoked child-operation is not one that the invoked parent-operation referred to by the linked-identifier allows.
Return-result-problem:		
–	unrecognizedInvocation	signifies that no operation with the specified invoke-identifier is in progress.
–	resultResponseUnexpected	signifies that the invoke operation does not report a result.
–	mistypedResult	signifies that the type of the result parameter supplied is not expected.
Return-error-problem:		
–	unrecognizedInvocation	signifies that no operation with the specified invoke-identifier is in progress
–	errorResponseUnexpected	signifies that the invoked operation does not report failure
–	unrecognizedError	signifies that the reported error is not one expected.
–	unexpectedError	signifies that the reported error is not one that the invoked operation may report
–	mistypedParameter	signifies that the type of the error parameters supplied is not one that is expected.

.

Printed copies can be ordered from:

ECMA
114 Rue du Rhône
CH-1204 Geneva
Switzerland

Fax: +41 22 849.60.01
Internet: helpdesk@ecma.ch

Files can be downloaded from our FTP site, **ftp.ecma.ch,** logging in as **anonymous** and giving your E-mail address as **password**. This Standard is available from library **ECMA-ST** as a compacted, self-expanding file in MSWord 6.0 format (file E254-DOC.EXE) and as an Acrobat PDF file (file E254-PDF.PDF). File E254-EXP.TXT gives a short presentation of the Standard.

Our web site, http://www.ecma.ch, gives full information on ECMA, ECMA activities, ECMA Standards and Technical Reports.

ECMA

114 Rue du Rhône
CH-1204 Geneva
Switzerland

This Standard ECMA-254 is available free of charge in printed form and as a file.

See inside cover page for instructions.