The ATM Forum

Technical Committee

PNNI Addendum on PNNI/B-QSIG Interworking and Generic Functional Protocol for the Support of Supplementary Services

AF-CS-0102.000

October, 1998

© 1998 by The ATM Forum. The ATM Forum hereby grants its members the limited right to reproduce in whole, but not in part, this specification for its members internal use only and not for further distribution. This right shall not be, and is not, transferable. All other rights reserved. Except as expressly stated in this notice, no part of this document may be reproduced or transmitted in any form or by any means, or stored in any information storage and retrieval system, without the prior written permission of The ATM Forum.

The information in this publication is believed to be accurate as of its publication date. Such information is subject to change without notice and The ATM Forum is not responsible for any errors. The ATM Forum does not assume any responsibility to update or correct any information in this publication. Notwithstanding anything to the contrary, neither The ATM Forum nor the publisher make any representation or warranty, expressed or implied, concerning the completeness, accuracy, or applicability of any information contained in this publication. No liability of any kind shall be assumed by The ATM Forum or the publisher as a result of reliance upon any information contained in this publication.

The receipt or any use of this document or its contents does not in any way create by implication or otherwise:

- Any express or implied license or right to or under any ATM Forum member company's patent, copyright, trademark or trade secret rights which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
- Any warranty or representation that any ATM Forum member companies will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
- Any form of relationship between any ATM Forum member companies and the recipient or user of this document.

Implementation or use of specific ATM standards or recommendations and ATM Forum specifications will be voluntary, and no company shall agree or be obliged to implement them by virtue of participation in The ATM Forum.

The ATM Forum is a non-profit international organization accelerating industry cooperation on ATM technology. The ATM Forum does not, expressly or otherwise, endorse or promote any specific products or services.

NOTE: The user's attention is called to the possibility that implementation of the ATM interoperability specification contained herein may require use of an invention covered by patent rights held by ATM Forum Member companies or others. By publication of this ATM interoperability specification, no position is taken by The ATM Forum with respect to validity of any patent claims or of any patent rights related thereto or the ability to obtain the license to use such rights. ATM Forum Member companies agree to grant licenses under the relevant patents they own on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. For additional information contact:

The ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1-650-949-6700
Fax: +1-650-949-6705

CONTENTS

This addendum to PNNI 1.0 1996 “Private Network-Network Interface Specification Version 1.0” (af-pnni-0055.000) contains the description and specification of the following capabilities:

- interworking with networks employing B-QSIG;
- Generic support for Supplementary Services (GSS).

Interworking with networks employing B-QSIG is described in new subsection 4.9 and specified in new Annex J. New Annex K contains PNNI PICS proforma additions for this capability.

Generic support for Supplementary Services (GSS) is described in new subsection 4.10 and specified in new Annexes L, M, N and O. New Annex P contains PNNI PICS proforma additions for this capability. New Appendices J, K, L, M, N, O and P contain additional information relating to this capability. In addition, there are enhancements to the routing specification, section 5 and to the MIB, Annex H.

Annex G, Minimum Subsets, is extended to include these two new capabilities.

New subsections 4.9 and 4.10 and the new Annexes and Appendices are contained in Part 1 of this addendum. Enhancements to section 5, Annex G and Annex H are contained in Part 2 of this addendum.

PART 1: New subsections, Annexes and Appendices

4.9 Interworking with networks employing B-QSIG signalling

Annex J specifies optional procedures for interworking between PNNI and the B-QSIG signalling system.

B-QSIG basic call/connection control is specified in ISO/IEC 13247 (1997). It is a signalling protocol based on the PNNI signalling protocol but applied between nodes of private ATM networks that employ static hop-by-hop routing. Typically these are networks that have evolved from narrowband Private Integrated Services Networks (PISNs) based on Integrated Services Digital Network (ISDN) technology.

4.10 Generic functional protocol for the support of supplementary services (GSS)

Annex L provides optional signalling capabilities for the support of supplementary services. It contains generic mechanisms that can be used by future protocols that support specific supplementary services. Supplementary services add value to basic call/connection handling. The following are examples of supplementary services:

- call diversion - a supplementary service that allows a user’s incoming calls to be redirected to another destination address prior to answer;
- call transfer - a supplementary service that allows a user to redirect a call, after answer, to another user.

Supplementary services typically involve the exchange of signalling information across a network between Application Service Control (AS-Control) entities, which provide supplementary-service-specific functionality.

The main features of generic support for supplementary services are as follows:

- the Remote Operations Service Element (ROSE), which provides a formalised method of information exchange between two AS-Control entities independently of network considerations;
- Generic Functional Transport Control (GFT-Control), which provides a means of conveying ROSE information between AS-Control entities located in different nodes of the network via any intermediate nodes;
- Three separate transport mechanisms for conveying GFT-Control and ROSE information between adjacent nodes either in association with a particular call/connection (bearer-related) or independently of any call/connection (bearer-independent).

Figure 4-4 shows the protocol architecture based around these entities, as applied to communication between non-adjacent nodes.

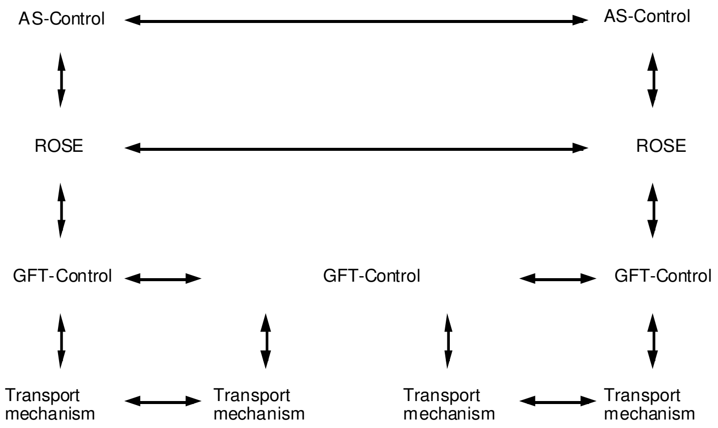


Figure 4-4: Protocol architecture, as applied to communication between non-adjacent nodes

ROSE, GFT-Control and the transport mechanisms are described in more detail below.

Generic support for supplementary services is based on the DSS2 generic functional protocol specified in ITU-T Recommendation Q.2932.1 but extended to allow non-local information exchange (via intermediate nodes) as well as local information exchange (between adjacent nodes).

4.10.1 ROSE

ROSE, as specified in ITU-T Recommendations X.219, X.229 and X.880, provides a formalised means of conducting information exchange ("operations") between application entities. Operations can be unconfirmed, confirmed only in the event of a positive outcome, confirmed only in the event of a negative outcome, or confirmed in either case. ROSE involves the use of four types of Application Protocol Data Units (APDUs):

- invoke APDU for initiating an operation;
- return result APDU for responding to an invoke APDU to indicate success;
- return error APDU for responding to an invoke APDU to indicate failure;
- reject APDU to report protocol problems.

4.10.2 GFT-Control

GFT-Control is based around a protocol element called the Network Facility Extension (NFE), which supports non-local information exchange by indicating (within the context of a connection - see 4.x.3) the source and destination of the ROSE APDU(s) it conveys. A destination node can be identified by one of the following means:

- explicitly using a node address, thereby causing the NFE and associated APDU(s) to be relayed on along the path of the connection until that particular node is reached;
- implicitly by indicating "end node", thereby causing the NFE and associated APDU(s) to be relayed on along the path of the connection until the node at the end of the connection is reached;
- implicitly by indicating "any node", thereby allowing any node that understands and can act upon the APDU(s) conveyed to intercept the information.

By omitting the NFE, information exchange is local.

4.10.3 Transport mechanisms

Information to be exchanged (ROSE APDUs and the NFE) are transported in the Facility information element.

For bearer-related transport, the Facility information element can be included in many of the messages defined in section 6 and also in the FACILITY message for transporting information at other times.

For bearer-independent transport, two mechanisms are specified.

The Connection-Oriented, Bearer-Independent (CO-BI) mechanism transports information within the context of a signalling connection established for the purpose. Establishment and release of a CO-BI signalling connection is similar to establishment and release of a call/connection, except that only a subset of the messages and information elements apply. However, a special message, CO-BI SETUP, is used instead of the SETUP message, thereby indicating that a CO-BI signalling connection is required. The Facility information element can be conveyed in most of the messages used for signalling connection establishment and release and also in the FACILITY message during the lifetime of an established signalling connection.

The Connectionless, Bearer-Independent (CL-BI) mechanism transports the Facility information element in a connectionless manner using the FACILITY message with a dummy call reference.

24. Annex J: Interworking with networks employing B-QSIG signalling (optional)

24.1 Scope

This Annex specifies optional procedures for interworking between PNNI and the B-QSIG signalling system.

B-QSIG basic call/connection control is specified in ISO/IEC 13247 (1997). It is a signalling protocol based on the PNNI signalling protocol but applied between nodes of private ATM networks that employ static hop-by-hop routing. Typically these are networks that have evolved from narrowband Private Integrated Services Networks (PISNs) based on Integrated Services Digital Network (ISDN) technology.

The procedures specified in this Annex apply to interworking between B-QSIG and PNNI.

24.2 References

ISO/IEC 13247 (1997): Information technology - Telecommunications and information exchange between systems - Broadband Private Integrated Services Network - Inter-exchange signalling protocol - Basic call/connection control

ISO/IEC 15773 (1997): Information technology - Telecommunications and information exchange between systems - Broadband Private Integrated Services Network - Inter-exchange signalling protocol - Transit counter additional network feature.

24.3 Definitions

gateway node: a node that provides interworking between PNNI and B-QSIG.

24.4 Acronyms

AINI	ATM Inter-Network Interface
CCH	Call/Connection Handling
PISN	Private Integrated Services Network

24.5 General

24.5.1 Intra-network Interworking

For the case of interworking between B-QSIG and PNNI inside a particular network, interworking is provided by a gateway node between the part of the network employing PNNI and the part employing B-QSIG (see figure 24-1). The B-QSIG part is regarded as an exterior link from the PNNI gateway node. The architectural model, the signalling model, and the detailed interworking specifications for this case are provided in section X.5 below. In section X.5, the PNNI part of the network is referred to as the “PNNI network” and the B-QSIG part of the network is referred to as the “B-QSIG network”.

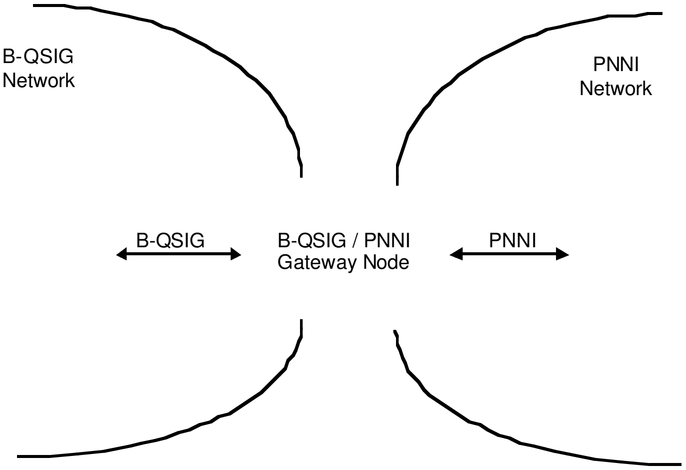


Figure 24-1: Architectural reference model for intra-network interworking

24.5.2 Inter-network Interworking

A B-QSIG network and PNNI network may be connected together by an inter-network interface connecting a B-QSIG gateway node to a PNNI border node. The inter-network interface is regarded as an exterior link from the PNNI border node.

> *NOTE. The ATM Forum plans to publish an inter-network interface known as the ATM Inter-Network Interface (AINI), which will be suitable for this and other applications. The architectural model for inter-network interworking using AINI is shown in figure 24-2. Further details are outside the scope of this Annex.*

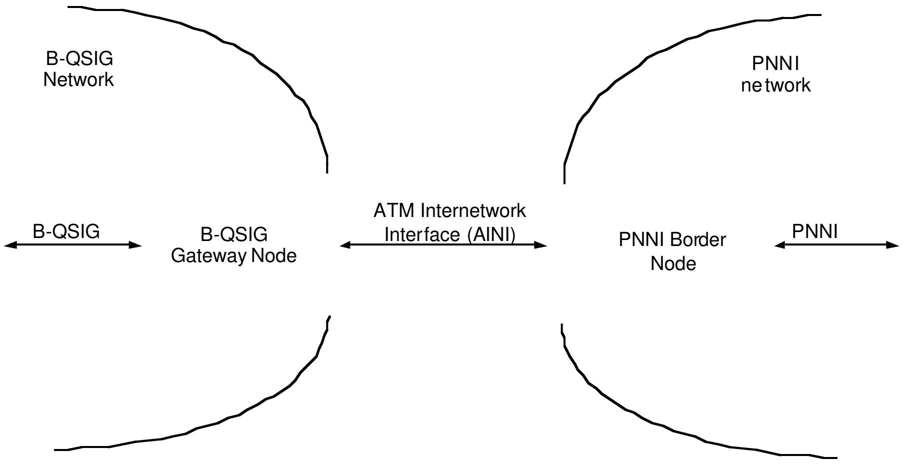


Figure 24-2: Architectural reference model for inter-network interworking

24.6 Intra-network Interworking

24.6.1 Basic Model

24.6.1.1 Architectural reference model

Figure 24-3 is a reference model for a node that acts as a gateway between a network employing PNNI and a network employing B-QSIG. Such a node will have one or more interfaces that use PNNI signalling and routing and one or more interfaces that use B-QSIG signalling.

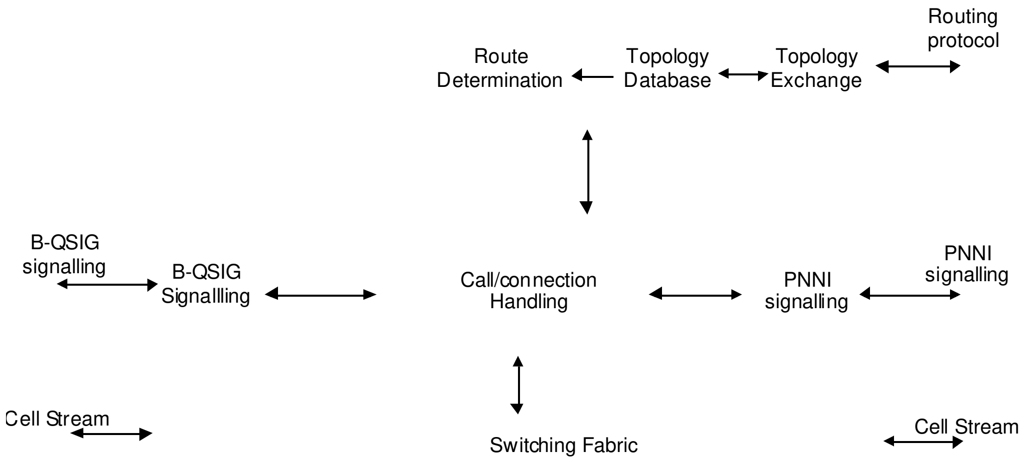


Figure 24-3: Architectural reference model for a B-QSIG/PNNI gateway node

NOTE. Figure 24-3 does not show UNI signalling, which is outside the scope of this specification.

24.6.1.2 Signalling protocol model

Figure 24-4 shows the signalling protocol model for B-QSIG/PNNI interworking at a gateway node. Call/Connection Handling (CCH) provides mediation between the two signalling protocols. CCH gives the appearance of a B-QSIG CCH entity to the next B-QSIG node and of a PNNI Call Control entity to the next PNNI node.

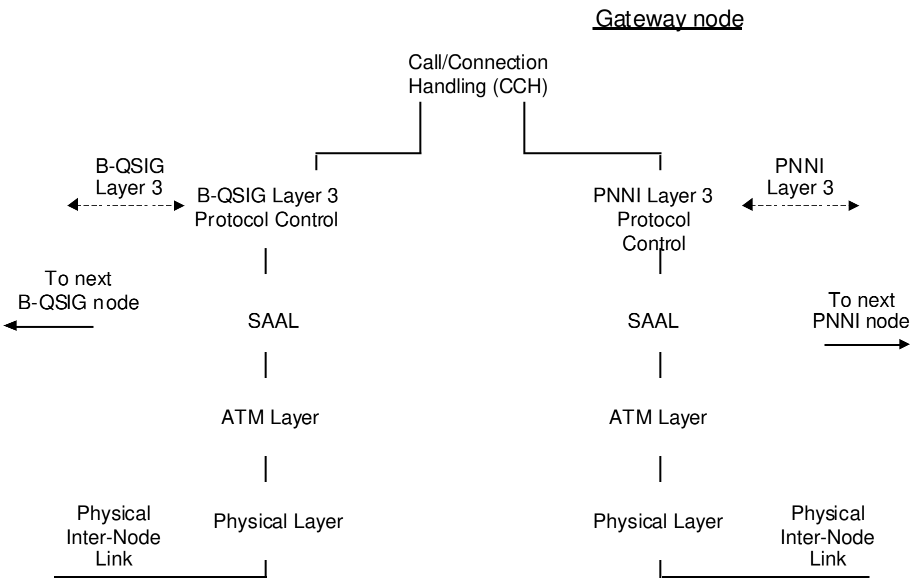


Figure 24-4: Signalling protocol model for B-QSIG/PNNI interworking

24.6.1.3 Interworking with other networks beyond the B-QSIG network

B-QSIG provides for interworking with other networks, in particular public B-ISDNs via DSS2, public narrow-band networks via DSS1, and private narrow-band networks using QSIG. This specification therefore also allows interworking between a PNNI network and any of these other networks, via a B-QSIG network.

24.6.2 Routing interworking

B-QSIG currently relies on hop-by-hop routing and the manual configuration of individual nodes to achieve this. Therefore the gateway node is required to provide interworking between a PNNI routing domain and the B-QSIG network.

The method of working is for the gateway node to operate PNNI routing protocols on PNNI interfaces and no routing protocol on B-QSIG interfaces, which are treated as exterior links. The gateway node must be configured with a list of exterior reachable addresses reachable via the B-QSIG network. This is in addition to any internal reachable addresses associated with end systems attached directly to the gateway node.

24.6.3 Signalling interworking for basic call/connection control

Signalling interworking between a B-QSIG link and a PNNI link is carried out within the Call/Connection Handling (CCH) entity in the gateway node. It is the same as interworking between two PNNI links with the following exceptions.

24.6.3.1 Handling of the Designated transit list information element

Because B-QSIG uses hop-by-hop routing, it does not use the Designated transit list information element.

24.6.3.1.1 Call/connection or party establishment in the direction PNNI to B-QSIG

For a SETUP or ADD PARTY message in the direction PNNI to B-QSIG, all Designated transit list information elements should have been exhausted by the time the gateway node is reached. The B-QSIG SETUP or ADD PARTY message shall contain no Designated transit list information element.

24.6.3.1.2 Call/connection or party establishment in the direction B-QSIG to PNNI

A SETUP or ADD PARTY message in the direction B-QSIG to PNNI will contain no Designated transit list information elements. The gateway node shall act as DTL originator by building its own stack of DTLs and including them in Designated transit list information elements in the PNNI SETUP or ADD PARTY message.

24.6.3.2 Handling of the Crankback information element

Because B-QSIG uses hop-by-hop routing, it does not use the Crankback information element.

24.6.3.2.1 Call/connection or party establishment in the direction PNNI to B-QSIG

A RELEASE, RELEASE COMPLETE or ADD PARTY REJECT message from the B-QSIG link during call/connection or party establishment will not contain a Crankback information element. It is an implementation matter for the gateway node whether to:

- reroute the call/connection or party establishment to another exterior link; or
- reroute the call/connection or party establishment within the PNNI routing domain and include one or more Designated transit list information elements, provided the gateway node is an entry border node or DTL originator; or
- pass back the indication of failure by means of a PNNI RELEASE or ADD PARTY REJECT message.

In the last case, the gateway node shall include a Crankback information element when any of the conditions for crankback as specified in Annex B apply.

24.6.3.2.2 Call/connection or party establishment in the direction B-QSIG to PNNI

A RELEASE, RELEASE COMPLETE or ADD PARTY REJECT message from the PNNI link during call/connection or party establishment can contain a Crankback information element. In this case the gateway node is acting as DTL originator. Depending on the contents of the Crankback information element and other factors, the gateway node may either:

- reroute the call/connection or party establishment within the PNNI routing domain; or
- reroute the call/connection or party establishment outside the PNNI routing domain, i.e., to an exterior link different from the B-QSIG link on which the call/connection or party establishment arrived; or
- pass back the indication of failure by means of a B-QSIG RELEASE or ADD PARTY REJECT message.

24.6.3.3 Handling of additional B-QSIG progress descriptions

Within the Progress indicator information element, B-QSIG supports additional progress descriptions 16, 17, 18, 19 and 21, which are not defined in PNNI. These progress descriptions use the value "01" (ISO/IEC standard) in the coding standard field.

In the case of a call/connection supporting a 64 kbit/s-based ISDN circuit mode service, any of these five additional progress descriptions can be received from a B-QSIG link in any message in which the Progress indication information element is permitted. Also the number of appearances of the Progress indicator information element in a B-QSIG message can exceed the maximum number of appearances of the Progress indicator information element in a PNNI message. A B-QSIG Progress indicator information element that contains a progress description not supported in PNNI or that would break the PNNI limit for the number of Progress indicator information elements in the message concerned shall be included in the corresponding PNNI message with the IE action indicator field containing the value "discard information element and proceed" and the pass along request field containing the value "pass along request".

> *NOTE. The purpose of passing the information element on is that a further B-QSIG network may be encountered further along the path of the call/connection and may be able to use the information concerned.*

In the case of any other call/connection, additional progress descriptions 16 and 21 can be received from a B-QSIG link in a Progress indicator information element in a PROGRESS, SETUP, ALERTING or CONNECT message. A B-QSIG PROGRESS message shall be passed on to the PNNI link with the message action indicator field containing the value "discard and ignore" and the pass along request field containing the value "pass along request". A B-QSIG Progress indicator information element that contains progress description 16 or 21 shall be included in the corresponding PNNI message with the IE action indicator field containing the value "discard information element and proceed" and the pass along request field containing the value "pass along request".

> *NOTE. The purpose of passing the PROGRESS message and/or information element on is that a further B-QSIG network may be encountered further along the path of the call/connection and may be able to use the information concerned.*

24.6.3.4 Handling of overlap sending in B-QSIG (support of 64 kbit/s-based circuit mode services only)

If a SETUP message from the B-QSIG link contains an incomplete number in the Called party number information element or a number that the gateway node cannot determine to be complete, the gateway node shall await further called party number information (in INFORMATION messages) until it determines that the number is complete before sending a SETUP message on the PNNI link. The gateway node shall not send a PNNI SETUP message containing a number that is known to be incomplete. A B-QSIG INFORMATION message shall not be passed on to a PNNI link.

> *NOTE. A gateway node can determine that a number is complete by analysis of the received digits or if it receives a Broadband sending complete information element in the SETUP message or an INFORMATION message. In the event of failure to receive a further INFORMATION message within a certain time (B-QSIG timer T302), the gateway can assume that the number is complete unless by analysis it knows that the number is incomplete.*

24.6.3.5 Handling of B-QSIG Broadband sending complete information element

If the Broadband sending complete information element is received in a B-QSIG SETUP message, the information element shall be omitted from the SETUP message passed on to the PNNI link.

24.6.3.6 Handling of number formats not supported in PNNI in the Called party number information element

If a number format (as indicated by the type of number and numbering plan identification fields) that is not supported by PNNI is received in a B-QSIG Called party number information element and the number can be mapped algorithmically or by database look-up into a format that is supported by PNNI, the mapped number shall be included in the PNNI Called party number information element. If the number cannot be mapped, the call/connection establishment or party establishment shall not be allowed to proceed into the PNNI network.

24.6.3.7 Handling of number formats not supported in PNNI in the Calling party number and Connected number information elements

If a number format (as indicated by the type of number and numbering plan identification fields) that is not supported by PNNI is received in a B-QSIG Calling party number or Connected number information element and the number can be mapped algorithmically or by database look-up into a format that is supported by PNNI, the mapped number shall be included in the corresponding PNNI information element. If the number cannot be mapped, the corresponding PNNI information element shall not include a number and shall have value "number not available" in the presentation indicator field.

24.6.3.8 Handling of the B-QSIG OAM traffic descriptor information element

If an OAM traffic descriptor information element is received in a B-QSIG SETUP or CONNECT message, the information element shall be included in the corresponding PNNI message with the IE action indicator field containing the value "discard information element and proceed" and the pass along request field containing the value "pass along request".

> *NOTE. The purpose of passing the information element on is that a further B-QSIG network may be encountered further along the path of the call/connection and may be able to use the information concerned.*

24.6.3.9 Handling of B-QSIG information elements from codesets other than codeset 0

If a B-QSIG information element from a codeset other than codeset 0 is received in a message that is to be passed on to the PNNI link, the information element shall be included in the PNNI message with the IE action indicator field containing the value "discard information element and proceed" and the pass along request field containing the value "pass along request". The corresponding Broadband locking shift or Broadband non-locking shift information element shall also be passed on.

> *NOTE. The purpose of passing the information element on is that a further B-QSIG network may be encountered further along the path of the call/connection and may be able to use the information concerned.*

In the case of the Transit counter information element, as specified in ISO/IEC 15773, the gateway node shall increase the transit counter value by at least one to take account of nodes expected to be passed through in the PNNI network. The method of calculating the amount of increase is implementation dependent.

25. Annex K: Protocol Implementation Conformance Statement (PICS) proforma for B-QSIG interworking[1]

This Annex is applicable only if optional Annex J is supported.

This Annex provides the PICS proforma for B-QSIG interworking. It is an addition to the PNNI PICS proforma, as specified in ATM Forum PNNI v1.0 Errata and PICS (af-pnni-81.00, March 1997).

The supplier of a protocol implementation which is claimed to conform to PNNI version 1.0 and also to B-QSIG interworking, as specified in Annex J, is required to complete a copy of the PICS proforma provided in 25.1 and attach it to the completed copy of the PICS proforma for PNNI 1.0.

25.1 B-QSIG interworking

Table 25-1

Item	Does the implementation support...	Conditions for status	Status	Reference	Support
Q1	Interworking with networks employing B-QSIG signalling?		O	Annex J (Section 24)	[ ]Yes [ ]No
Q2	Routing interworking?	Q1 NOT Q1	M N/A	24.6.2	[ ]Yes [ ]N/A
Q3	Signalling interworking for basic call/connection control?	Q1 NOT Q1	M N/A	24.6.3	[ ]Yes [ ]N/A
Comments:					

---

[1] **Copyright release for PICS:** This PICS proforma may be freely reproduced, so that it may be used for its intended purpose.

26. Annex L: Generic Functional Protocol for the Support of Supplementary Services (GSS) (Optional)

26.1 Scope

This Annex, together with Annexes M, N and O, specifies the generic functional protocol for the support of supplementary services and additional basic call capabilities across a PNNI. It forms an optional addition to the PNNI signalling protocol.

The procedures specified in this Annex 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 Annex to individual additional basic call capabilities and supplementary services is outside the scope of this implementation agreement and should be defined as part of the specification of such individual capabilities.

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

26.2 Normative references

ITU-T Recommendation E.164: 1991, *Numbering plan for the ISDN era.*

ITU-T Recommendation Q.2932.1: 1996, *B-ISDN Digital Subscriber Signalling System No. 2 (DSS2); Generic Functional Protocol Part 1.*

ITU-T Recommendation X.208: 1993, *Specification of Abstract syntax Notation One (ASN.1).*

ITU-T Recommendation X.209: 1993, *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*

ITU-T Recommendation X.213: 1992, *Information technology – Network service definition for open systems interconnection.*

ITU-T Recommendation X.229: 1993, *Remote Operations: Protocol Specification.*

ITU-T Recommendation X.249, *Remote Operations Service Element - Protocol Implementation Conformance Statement (PICS) proforma.*

ITU-T Recommendation X.680: 1994, *Information Technology - Abstract Syntax Notation One (ASN.1): Specification of Basic Notation.*

ITU-T Recommendation X.690: 1994, *Information Technology - ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rues (CER) and Distinguished Encoding Rules (DER).*

ITU-T Recommendation X.880: 1995, *Information Technology - Remote operations, Concepts, model and notation.*

26.3 Definitions

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

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

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

End node: In the context of a particular call/connection or a particular CO-BI connection, an Originating or Terminating node, or node that provides interworking with another signalling system without transporting APDUs unchanged to and from that other signalling system.

Next node: An adjacent node 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 node: In the context of a call/connection, the node to which the calling endstation is attached.

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

Terminating node: In the context of a call/connection, the node to which the called endstation is attached.

Transit node: In the context of a call/connection or CO-BI connection, a node at which the connection transits from one PNNI link to another PNNI link.

26.4 Abbreviations

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

GSS	Generic support for Supplementary Services
MSI	Manufacturer Specific Information
PICS	Protocol Implementation Conformance Statement

26.5 Description

26.5.1 Overview

The generic functional protocol provides a means of exchanging ROSE APDUs on behalf of Application Service Control entities located in different nodes. 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 section 6 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 specification.

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

26.5.2 Addressing mechanisms

Communication between adjacent nodes does not require addressing. Where the nodes 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.

26.5.2.1 Explicit addressing

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

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

The assigned number is according to the numbering plan of the PNNI network.

26.5.2.2 Functional addressing

The following functions are provided:

- End node;
- any type of node (i.e. the next node that understands the contents).

26.5.3 Protocol architecture

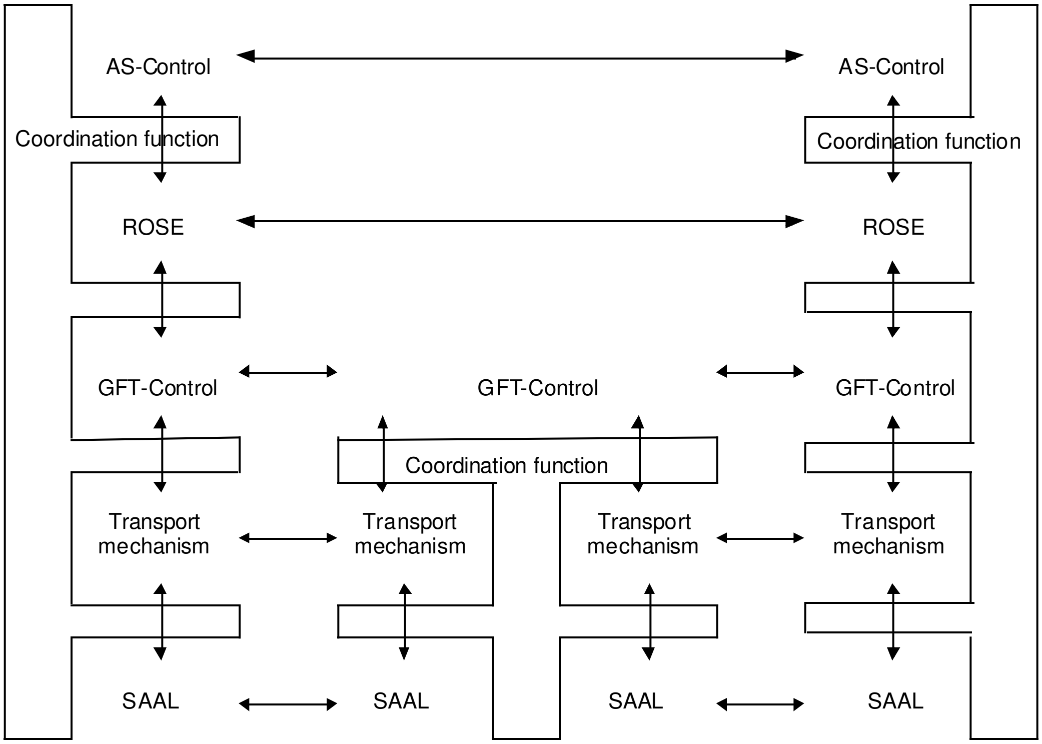
Protocol Architecture described in ITU-T Recommendation Q.2932.1 clause 5.2 shall apply with following modification:

- Replace all references to ITU-T Recommendation Q.2931 and Q.2971 with references to PNNI.

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

Figure 26-1 shows the application of the protocol model to the case where the AS-Control entities to be associated in communication are not in adjacent nodes. In the example shown, communication is via a single intervening node. It may be generally applied to communication via more than one intervening node by simple replication.

In Figure 26-1, relaying functions at the intervening node are performed by GFT-Control



* The transport mechanism can be CO-BR, CO-BI or CL-BI

Figure 26-1: Application of the protocol model to communication between non-adjacent nodes

26.5.4 Services provided by individual protocol entities

26.5.4.1 Services provided by ROSE

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

26.5.4.2 Services provided by GFT-control

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

26.5.4.3 Services provided by bearer-related transport

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

- Replace all references to ITU-T Recommendation Q.2931 and Q.2971 with references to PNNI.

26.5.4.4 Services provided by connectionless bearer-independent transport

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

26.5.4.5 Services provided by connection-oriented bearer-independent transport

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

26.6 Operational requirements

The requirements for provision of this specification 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.

26.7 Primitive definitions and state definitions

26.7.1 Primitive definitions

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

- Replace all references to ITU-T Recommendation Q.2931 and Q.2971 with references to PNNI.

26.7.2 State definitions

26.7.2.1 APDU transport mechanisms

26.7.2.1.1 Bearer-related transport mechanism

There are no additional call/connection states over and above those defined in section 6.2.

26.7.2.1.2 Connectionless bearer-independent transport mechanism

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

26.7.2.1.3 Connection-oriented bearer-independent transport mechanism

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

26.7.2.2 GFT-Control

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

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

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

26.8 Coding requirements

26.8.1 Message functional definitions and content

This subsection shall be read in conjunction with 6.3. All messages are additional to those defined in 6.3 and the following tables should be interpreted according to the introductory material of 6.3.

To determine if an information element specified in this specification is allowed to be included in the following messages, see 26.8.2.

Information elements not defined in 26.8.2 are only allowed to be included in the following messages when explicitly indicated in the message structure.

26.8.1.1 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. In each case the maximum length and the maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit.

26.8.1.1.1 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 following modifications:

- the maximum length and maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit;
- the maximum length and the maximum number of repetitions of the Notification indicator information element are implementation options; and
- the Endpoint reference information element shall be included in the case of point-to-multipoint connections.

26.8.1.2 Messages for connectionless bearer-independent transport

26.8.1.2.1 FACILITY

This message may be sent between two nodes 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 26-1.

Table 26-1: FACILITY message content

Message type: FACILITY

Information element	**Reference**	**Type**	**Length**
Protocol discriminator	6.4.2	M	1
Call reference (NOTE 1)	6.4.3	M	4
Message type	6.4.4	M	2
Message length	6.4.4	M	2
Broadband repeat indicator	6.4.5.13	M (NOTE 2)	4-5
Facility (NOTE 3, NOTE 5)	25.9.2.2.2	M	10-*
Notification indicator	25.9.2.2.3	O (NOTE 4)	4-*
Called party number	6.4.5.15	M	4-*
Calling party number	6.4.5.17	M	4-*
Designated transit list (NOTE 6)	6.4.6.4	M	33-546
Connection scope selection	6.4.5.23	O (NOTE 7)	4-6

NOTE 1. The dummy call reference as specified in ITU-T Recommendation Q.2931 subclause 4.3 shall be used.

NOTE 2. When the Broadband repeat indicator information element immediately precedes the Designated transit list information element, it indicates the order of Designated transit list information elements in the DTL stack. This information element is mandatory, even when there is only one Designated transit list information element. It may also precede any other information element that is repeated.

NOTE 3. The maximum length and the maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit.

NOTE 4. This information element may be present whenever a notification is delivered. The Notification indicator information element may be repeated in this message. The maximum length and the number of repetitions of the Notification indicator information element are implementation options.

NOTE 5. The Network Facility Extension (NFE) field within the Facility information element shall be omitted.

NOTE 6. Included by the source node to indicate the hierarchical source route for the CL-BI message. Included by the node at the entry to a hierarchical level to indicate the path through that hierarchical level. This information element may be repeated up to 10 times.

NOTE 7. May be included when sending a CL-BI FACILITY message using a group *address in the Called party number information element.*

26.8.1.3 Messages for connection-oriented bearer-independent transport

26.8.1.3.1 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 6.3.1.2.

26.8.1.3.2 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 26-2.

Table 26-2: CO-BI SETUP message content

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

Information element	Reference	Type	Length
Protocol discriminator	6.4.2	M	1
Call reference	6.4.3	M	4
Message type	6.4.4	M	2
Message length	6.4.4	M	2
Broadband repeat indicator	6.4.5.13	M (NOTE 1)	4-5
Facility	26.8.2.2.2	O (NOTE 2)	4-*
Called party number	6.4.5.15	M	4-*
Calling party number	6.4.5.17	M	4-*
Designated transit list	6.4.6.4	M (NOTE 4)	33-546
Connection scope selection	6.4.5.23	O (NOTE 5)	4-6
Notification indicator	26.8.2.2.3	O (NOTE 3)	4-*

NOTE 1. When the Broadband repeat indicator information element immediately precedes the Designated transit list information element, it indicates the order of Designated transit list information elements in the DTL stack. This information element is mandatory, even when there is only one Designated transit list information element. It may also precede any other information element that is repeated.

NOTE 2. Included if the requesting GFT-Control wishes to include APDUs in the set up request. The maximum length and the maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit.

NOTE 3. This information element may be present whenever a 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.

NOTE 4. Included by the source node to indicate the hierarchical source route for the

CO-BI connection. Included by the node at the entry to a hierarchical level to indicate the path through that hierarchical level. This information element may be repeated up to 10 times

NOTE 5. May be included when establishing a CO-BI connection using a group address in the Called party number information element.

26.8.1.3.3 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 following :modifications:

- the maximum length and maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit; and
- the maximum length and the maximum number of repetition of the Notification indicator information element are implementation options.

26.8.1.3.4 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.

The maximum length and the maximum number of repetitions of the Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit. The maximum length and the maximum number of repetitions of the Notification indicator information element are implementation options.

26.8.1.3.5 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 6.3.1.9.

26.8.1.3.6 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 Facility information element are subject to the message length not exceeding the SSCOP/SSCF payload limit and the maximum length and maximum number of repetitions of the Notification indicator information element are implementation options. In addition, the Crankback information element may optionally be included as shown in table 26-3.

Table 26-3: RELEASE message additional content

Information element	Reference	Type	Length
Crankback	6.4.6.3	O	4-72

26.8.1.3.7 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 6.3.1.5.

26.8.1.3.8 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 6.3.1.7.

26.8.1.3.9 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 6.3.1.8.

26.8.2 General message format and information element coding

Section 6.4 shall apply with the following additions.

26.8.2.1 Message type

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

26.8.2.2 Other information elements

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

26.8.2.2.1 Call state

The call state information element is as defined in 6.4.5.14. However the state value assignments defined in Table 9/Q.2932.1 exist for the connection-oriented bearer-independent transport mechanism.

26.8.2.2.2 Facility

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

Figure 26-2 shows the structure of the Facility information element. Table 26-4 shows the value of the protocol profile field applicable for supplementary services or additional basic call capabilities.

<table>
<tr><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>Octets</th></tr>
<tr><td colspan="8">Facility</td><td></td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td colspan="8">Information element identifier</td><td></td></tr>
<tr><td>1</td><td colspan="2" rowspan="2">Coding standard</td><td colspan="5">Information element instruction field</td><td>2</td></tr>
<tr><td>ext</td><td>Flag</td><td>Pass along indicator</td><td colspan="3">Information element action indicator</td><td></td></tr>
<tr><td colspan="8">Length of facility contents</td><td>3</td></tr>
<tr><td colspan="8"></td><td>4</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td colspan="5" rowspan="2">Protocol profile</td><td>5</td></tr>
<tr><td>ext.</td><td colspan="2">Spare</td><td></td></tr>
<tr><td colspan="8">Network Facility Extension (NFE) (NOTE 1)</td><td>5.1 *
(NOTE 2,,
NOTE 5)</td></tr>
<tr><td colspan="8">Interpretation APDU (NOTE 3)</td><td>5.2 *
(NOTE 2,
NOTE 5)</td></tr>
<tr><td colspan="8">ROSE APDU(s) (NOTE 4)</td><td>6 etc.
(NOTE 5)</td></tr>
</table>

NOTE 1. The Network Facility Extension (NFE), as defined in 26.8.2.2.2.1, may be included, in accordance with the procedures of 26.9.2.

NOTE 2. Each of octets groups 5.1 and 5.2 comprises an ASN.1 type encoded as defined 26.8.2.3. 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 26.8.2.2.2.2, may be included, in accordance with the procedures in 26.9.2.

NOTE 4. One or more ROSE APDUs in accordance with 26.8.2.2.2.3 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 26-2: Facility information element

Table 26-4: Facility information element protocol profile

Bits

5 4 3 2 1

1 1 1 1 1 Networking extensions (NOTE)

All other values are reserved and their usage is the subject of other specifications

NOTE. ISO/IEC defined local values apply in the ROSE APDUs. These local values are specified in ISO/IEC standards using this specification.

The remainder of the contents of the Facility information element is specified in the following subsections using ASN.1. Each ASN.1 module is specified twice: once using the old ASN.1 notation as specified in ITU-T Recommendation X.208, and once using the new ASN.1 notation as specified in ITU-T Recommendation X.680. The use of the two different abstract syntaxes allows the use of compilers for either syntax. Also, although the new syntax should be the preferred syntax for new supplementary services specifications, the inclusion in this Specification of ASN.1 modules using the old syntax facilitates the adoption of existing supplementary services specifications that use the old syntax.

Both notations result in the same transfer syntax when encoded using the Basic Encoding Rules (see 26.8.2.3), and therefore there is full compatibility between implementations compiled from the old abstract syntax and implementations compiled from the new abstract syntax.

26.8.2.2.2.1 Network Facility Extension (NFE)

The NFE shall comprise ASN.1 type NetworkFacilityExtension as defined in 27.3 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 and in 28.3 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680, encoded in accordance with 26.8.2.3. 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 PNNI network, and a means of identifying the origin and destination of the information, in accordance with the procedure of 26.9.2.

Subsection describes the use of the particular elements of NFE.

26.8.2.2.2.2 Interpretation APDU

The Interpretation APDU shall comprise ASN.1 type InterpretationAPDU as defined in 27.6 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 and in 28.6 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680, encoded in accordance with 26.8.2.3. This APDU provides a means whereby the originator can include optional instructions to the receiving node 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.

Subsection 26.9.2 describes the use of the Interpretation APDU.

26.8.2.2.2.3 ROSE APDU

A ROSE APDU shall comprise ASN.1 type APDU as defined in 27.1 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 and in 28.1 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680, encoded in accordance with 26.8.2.3.

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. The definition of types InvokeAPDU, ReturnResultAPDU, ReturnErrorAPDU and RejectAPDU in 27.1 and types Invoke, ReturnResult, ReturnError and Reject in 28.1 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 27.1 and 28.1 (e.g. InvokeIdType) are size delimited to enhance interoperability in a multi-vendor PNNI network.

NOTE. Appendix O gives a general overview of the ROSE protocol and its constituent parts. Appendix P 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).

26.8.2.2.3 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 26-3 and table 26-5, this being an extension of the coding specified in 6.4.5.27.

The maximum length of the information element is application dependent.

<table>
<tr><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>Octets</th></tr>
<tr><td colspan="8">Notification Indicator</td><td></td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td colspan="8">Information element identifier</td><td></td></tr>
<tr><td rowspan="2">1
Ext</td><td colspan="2" rowspan="2">Coding standard</td><td colspan="5">Information element instruction field</td><td rowspan="2">2</td></tr>
<tr><td>Flag</td><td>Pass along indicator.</td><td colspan="3">Information element action indicator</td></tr>
<tr><td colspan="8">Length of notification element contents</td><td>3</td></tr>
<tr><td colspan="8"></td><td>4</td></tr>
<tr><td>1</td><td colspan="7">Notification Description Encoding</td><td>5</td></tr>
<tr><td colspan="8">ASN.1 encoded Notification Data Structure</td><td>5.1
(NOTE)</td></tr>
</table>

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 26-3: Notification indicator information element

Table 26-5: 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 27.4 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 or using the NOTIFICATION object class specified in 28.4 of Annex N 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 26.12.3). 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 27.7 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 and in 28.7 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680, encoded in accordance with 26.8.2.3. Element notificationTypeID shall contain the notification value and element notificationArgument shall contain any additional data.

27.4 of Annex M and 28.4 of Annex N also define the ASN.1-encoded notification pnniIeNotification, which can be used to convey PNNI information elements as a notification. Other notifications will be defined using the NOTIFICATION macro in the relevant supplementary specifications (standardised or manufacturer specific).

26.8.2.2.4 Treatment of existing PNNI 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 PNNI information elements.

New parameters shall be defined using ITU-T Recommendation X.209 coding, or ITU-T Recommendation X.690 as appropriate, if they do not appear elsewhere in PNNI messages.

Supplementary service or additional basic call capability protocol specifies may elect to encapsulate one or more existing PNNI information elements within an ITU-T Recommendation X.209 data element, or ITU-T Recommendation X.690 data element, as appropriate, thereby retaining the coding specified in 6.4 for these information elements. When this option is chosen, all the PNNI information elements should be grouped together as the content following the PnniInformationElement 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 PnniInformationElement is defined in 27.2 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 and in 28.2 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680.

26.8.2.3 Encoding of information described using ASN.1

When specified according to ITU-T Recommendation 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 Recommendation X.209.

When specified according to ITU-T Recommendation 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 Recommendation 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.

26.9 Signalling procedures

26.9.1 APDU transport mechanisms

The transport function for operations is performed by the exchange of APDUs via PNNI signalling 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 connection-oriented bearer-independent call reference or, in the case of CL-BI, the dummy call reference.

26.9.1.1 Bearer-related transport

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

The procedures for call/connection control are described in 6.5 and 6.6. These procedures are not influenced by the APDUs carried. Bearer-related transport procedures and operations shall operate independently of each other.

26.9.1.1.1 Normal operation

For bearer-related transport any message in which the Facility information element may be included (see 26.8.1.1) 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 instruction indicator in the Facility information element shall be set as indicated by GFT-Control. If the message to be used is the FACILITY message, the message pass along request flag shall have the same value as the information element pass along request flag and the message action indicator shall be derived from the information element action field in accordance with table 26-6.

Table 26-6: Derivation of FACILITY message action indicator from Facility IE action indicator

Information element action indicator	Message action indicator
clear call	clear call
discard information element and proceed	discard and ignore
discard information element, proceed and report status	discard and report status
discard message and ignore	discard and ignore
discard message and report status	discard and report status

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 subsection 26.9.2) shall also apply.

26.9.1.1.2 Exceptional procedures

If a receiving entity recognizes 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 6.5 or 6.6, 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 specifications.

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 6.5 or 6.6, 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 26.9.1.1 are an extension to the procedures of 6.5 and 6.6. As such the general error handling procedures as defined in 6.5.6 apply. However, the handling of errors in octets 5 onwards of the Facility information element is specified in 26.9.2.1. The handling of errors in APDUs is specified in 26.9.3. If the connection is being cleared, the treatment of outstanding supplementary service or additional basic call capability requests is subject to the specifications for the individual supplementary services or additional basic call capabilities.

26.9.1.2 Bearer-independent transport mechanisms

Bearer-independent transport mechanism described in ITU-T Recommendation Q.2932.1 clause 9.1.2 shall apply.

26.9.1.3 Connection-oriented bearer-independent transport mechanism

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

26.9.1.3.1 Actions in the Null state

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

- Replace all references to ITU-T Recommendation Q.2931 with references to PNNI.
- Before a bearer-independent signalling connection can be established, a reliable signalling AAL connection shall be established, if not already available, using the procedures of 6.5.1. This replaces item a) of Q.2932.1 section 9.1.3.1.
- The preceding side shall include one or more Designated transit list information elements in the CO-BI SETUP message in accordance with the instructions from GFT-Control.
- When entering the call present state, a CALL PROCEEDING message shall be sent.

26.9.1.3.2 Actions in the Call Present state

Actions in the call present state described in ITU-T Recommendation 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.

26.9.1.3.3 Actions in the Call Initiated state

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

26.9.1.3.4 Actions in the Incoming Call Proceeding state

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

26.9.1.3.5 Actions in the Outgoing Call Proceeding state

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

26.9.1.3.6 Actions in the Active state

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

26.9.1.3.7 Connection release

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

26.9.1.3.8 Actions in the Release Request state

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

26.9.1.3.9 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 Recommendation Q.2932.1 clause 9.1.3.9 shall apply.

26.9.1.3.10 Protocol error handling

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

- Replace all references to ITU-T Recommendation Q.2931 with references to PNNI.

26.9.1.4 Connectionless bearer-independent transport mechanism

26.9.1.4.1 Normal operation

Before generic functional data can be sent, a reliable signalling AAL connection shall be established, if not already available, using the procedures of 6.5.1.

The connectionless transport mechanism is based on the FACILITY message as defined in 26.8.1.2.1.

When indicated by GFT control, generic functional data shall be included in one or more Facility information elements and transferred in a FACILITY message.

The sending node shall include one or more Designated transit list information elements in the FACILITY message in accordance with the instructions from GFT control.

On receipt of a valid FACILITY message containing the dummy call reference, the transport mechanism shall pass all valid received generic functional data and protocol profile values in the Facility information element to GFT-control.

26.9.1.4.2 Exceptional procedure

If a FACILITY message or Facility information element is received with an instruction indicator indicating "clear call" or "discard and report status", then this shall be treated as if "discard and ignore" had been received.

If a FACILITY message is received and it does not contain the Facility information element, the receiving entity shall discard the FACILITY message.

If a FACILITY message is received that contains a Facility information element with invalid content in octets 1 to 4, then that Facility information element and the FACILITY message shall be ignored and no action taken on the contents of the FACILITY message.

If a FACILITY message is received that contains an unexpected information element, or an optional information element with content error, then that information element shall be ignored, and action taken on the message and those information elements that are recognized and have valid content.

26.9.2 GFT-Control procedures for APDUs

26.9.2.1 GFT-control procedures for transport of APDUs

26.9.2.1.1 Actions at a source node

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.

For connection-oriented transport mechanisms, APDUs may be of two basic types:

- those which have only link significance, i.e. over a single link between two adjacent nodes (local information exchange) ; or,
- those which have network significance, between two nodes in the PNNI network which are not necessarily adjacent, and which can be, but need not be, the end nodes involved in the connection (non-local information exchange).

For local information exchange using connection-oriented transport mechanisms, the Network Facility Extension (NFE), defined in 26.8.2.2.2.1, shall not be included in the generic functional data.

For non-local information exchange using connection-oriented transport mechanisms, the NFE shall be included, encoded as described in table 26-7.

For connectionless transport, the NFE shall not be included.

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 node. Any relationship between such APDUs is beyond the scope of this specification.

Table 26-7: 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 node to End node	endNode (NOTE)	Not included	endNode	Not included
2	End node to addressed node	endNode (NOTE)	Not included	anyTypeOfNode	node address
3	End node to next node that understands contents	endNode (NOTE)	Not included	anyTypeOfNode	Not included
4	Transit node to End node	AnyTypeOfNode	node address	endNode	Not included
5	Transit node to addressed node	AnyTypeOfNode	node address	anyTypeOfNode	node address
6	Transit node to next node that understands contents	AnyTypeOfNode	node address	anyTypeOfNode	Not included

NOTE. The value endNode for the sourceEntity should be avoided if there is a possibility that the node can cease to be an End node (e.g., through the use of certain supplementary services) prior to a response (e.g., a Reject APDU) being received.

If a Source node wishes to include additional information to facilitate handling of unrecognized ROSE APDUs of type InvokeAPDU at a Destination node, it shall include an Interpretation APDU (see 26.8.2.2.2.2) 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.

26.9.2.1.2 Actions at a receiving node

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

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

26.9.2.1.2.1 End node actions

If the receiving node is an End node, and the encoding of the received NFE complies with the encoding and structure defined in 26.8.2.2.2.1, the following actions shall apply:

- if the destinationEntity element of the NFE indicates endNode or anyTypeOfNode and no destinationEntityAddress element is included, it shall become the Destination node for that generic functional data;
- if the destinationEntity element of the NFE indicates anyTypeOfNode and the NFE includes a destinationEntityAddress element containing an address that matches the node's own address, the node shall become the Destination node for that generic functional data;
- if the destinationEntity element of the NFE indicates endNode and erroneously includes a destinationEntityAddress element, the node shall become the Destination node for that generic functional data;
- if the destinationEntity element of the NFE indicates a value in the range 2 to 11, the receiving node shall become the Destination node 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 26.8.2.2.2.1, the entire Facility information element shall be discarded.

26.9.2.1.2.2 Transit node actions

If the receiving node is a Transit node, and the encoding of the received NFE complies with the encoding and structure defined in 26.8.2.2.2.1, the following actions shall apply:

- if the destinationEntity element of the NFE indicates anyTypeOfNode and the NFE includes a destinationEntityAddress element containing an address that matches the node's own address, the node shall become the Destination node for that generic functional data;
- if the destinationEntity element of the NFE indicates anyTypeOfNode and no destinationEntityAddress element is included, the node may become the Destination node for that generic functional data if it understands the contents;
- if the destinationEntity element of the NFE indicates endNode and erroneously includes a destinationEntityAddress element, the node 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 endNode, and the Transit node is capable of acting as an End node for all services indicated in the generic functional data, it may become the Destination node 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 node will act as if it were an End node. This may occur, for example, when a node at a 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 node is capable of acting as an End node for all services indicated in the generic functional data, it may become the Destination node 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 node does not become the Destination node, the generic functional data and instruction indicator shall be passed on unchanged to the Next node.

If the received NFE does not conform to the encoding and structure defined in 26.8.2.2.2.1, the entire generic functional data shall be discarded and no generic functional data shall be passed on to the Next node.

26.9.2.1.3 Actions at a destination node

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 6.5.6 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 unrecognizedOperation the action taken shall depend on the contents of the Interpretation APDU as follows:

- if the Interpretation APDU indicates rejectUnrecognizedInvokePdu the ROSE APDU of type RejectAPDU shall be delivered to the destination indicated by ROSE;
- if the Interpretation APDU indicates clearCallIfAnyInvokePduNotRecognized 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 discardAnyUnrecognizedInvokePDU 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.

26.9.2.2 GFT-Control procedures for CO-BI connection control

26.9.2.2.1 Actions at an Originating node

26.9.2.2.1.1 Actions in the Originating_connection_idle state

When a request for establishment of a CO-BI connection to a remote node 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 node, and enter the Originating_connection_request state. GFT-Control shall request the inclusion of one or more Designated transit list information elements in accordance with Annex A procedures for an originating node. Routing shall be as for point-to-point call establishment and in computing a route, only nodes for which the nodal information flag “CO-BI transport supported” is set to 1 shall be selected.

26.9.2.2.1.2 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.

26.9.2.2.1.3 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.

26.9.2.2.2 Actions at a Transit node

If GFT-Control receives indication from the protocol entity of a received CO-BI SETUP message from the Preceding node, it shall examine the contents of the Called party number and DTL information elements. If, in accordance with Annex A, this node is the DTL terminator and the Called party number information element matches that of the receiving node, this node shall become a Terminating node and act in accordance with 26.9.2.2.3. Otherwise it shall follow the procedures of the subsections below.

26.9.2.2.2.1 Actions in the Transit_connection_idle state

If the contents of the Called party number information element and DTL information element(s) contained in the CO-BI SETUP message indicate that this is not the destination node and if onward routing of the message is possible, GFT-Control shall request the protocol entity to send a CO-BI SETUP message on the appropriate inter-node link to the Succeeding node, associate the incoming and outgoing connections and enter the Transit_connection_request state. The handling of the DTL information element(s) shall be in accordance with Annex A. Routing shall be as for point-to-point call establishment. When carrying out further route computation, an entry border node shall select only nodes for which the nodal information flag “CO-BI transport supported” is set to 1.

If onward routing is not possible and this is not the destination node, GFT-Control shall request the protocol entity to release the connection by sending a RELEASE message to the Preceding node and remaining in the Transit_connection_idle state. A Crankback information element shall be included with the request to send a RELEASE message when any of the conditions for crankback as specified in Annex B apply.

26.9.2.2.2.2 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 node, GFT-Control shall start timer T310.

When the protocol entity informs GFT-Control of a CONNECT message received from the Succeeding node, GFT-Control shall request the protocol entity to send a CONNECT message to the Preceding node, 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 node, GFT-Control shall request the protocol entity to send a RELEASE message to the Preceding node, stop timer T310 if running and enter the Transit_connection_idle state. Alternatively, if the received message contains a Crankback information element, an entry border node may attempt rerouting in accordance with Annex B.

When the protocol entity informs GFT-Control that a RELEASE message has been received from the Preceding node, GFT-Control shall request the protocol entity to send a RELEASE message to the Succeeding node, 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 node and request the protocol entity to send a RELEASE message to the Succeeding node.

26.9.2.2.2.3 Actions in the Transit_connection_active state

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

26.9.2.2.3 Actions at a Terminating node

26.9.2.2.3.1 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 node, 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.

26.9.2.2.3.2 Actions in the Incoming_connection_active state

If the protocol entity informs GFT-Control that a RELEASE message has been received from the Preceding node, 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.

26.9.2.3 GFT-Control procedures for CL-BI mode

26.9.2.3.1 Actions at a source node

On receipt of a request to send APDUs using CL-BI transport, accompanied by the address of the destination node, GFT-Control shall:

- if a route to the destination can be selected, select the appropriate PNNI link based on the destination address given in the request and inform Protocol Control to send a FACILITY message which shall contain:
  - a Calling party number information element, identifying the address of the source node;
  - a Called party number information element, identifying the address of the destination node;
  - one or more Designated transit list information elements, identifying the hierarchical source route for the CL-BI message;
  - a Facility information element which shall not contain an NFE.
- if no route to the destination node can be selected, ignore the request.

Routing shall be as for point-to-point call establishment and in computing a route, only nodes for which the nodal information flag "CL-BI transport supported" is set to 1 shall be selected.

26.9.2.3.2 Actions at a receiving node

If a node receives a FACILITY message containing the dummy call reference on a PNNI link from an adjacent node, it shall examine the contents of the Called party number and DTL information elements. If, in accordance with Annex A, this node is the DTL terminator and the Called party number information element matches that of the receiving node, this node shall become the destination node and act in accordance with 26.9.2.3.3. If this is not the case, and the receiving node can route the FACILITY message based on the Called party number and the Designated transit list information element(s), the FACILITY message shall be sent on the appropriate PNNI link. The handling of the DTL information element(s) shall be in accordance with Annex A. Routing shall be as for point-to-point call establishment. When carrying out further route computation, an entry border node shall select only nodes for which the nodal information flag "CL-BI transport supported" is set to 1.

If the receiving node is not the destination node and onward routing of the FACILITY message is not possible, the receiving node shall discard the FACILITY message.

> *NOTE. It is the responsibility of the appropriate specification for the supplementary service utilising these transport procedures to ensure that the service can cope gracefully if the FACILITY message is discarded during routing.*

26.9.2.3.3 Actions at a destination node

If the received FACILITY message is destined for the receiving node, the contents of the Facility information element shall be treated in accordance with 26.9.2.1.3.

> *NOTE. It is the responsibility of SS-Control (i.e. the specific Supplementary service) in the destination node to store the Calling party number information element to enable response to the service request to be made using a further CL-BI message.*

If a received Facility information element contains an NFE, the receiving node shall ignore the contents of that NFE.

26.9.3 Remote operations procedures

26.9.3.1 Introduction

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

26.9.3.2 Procedures for operations

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

26.9.3.2.1 Invocation

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

26.9.3.2.2 Return result

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

26.9.3.2.3 Return error

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

26.9.3.2.4 Reject

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

26.9.3.2.5 Formal definition of data types

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

26.9.4 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.

26.9.4.1 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 connection 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 node; or
- if a SETUP or CO-BI SETUP message has been received from the Preceding node, but no response has been sent; or,

- if a clearing message has already been sent to or received from the Next node

the notification information shall be discarded.

No state change shall occur on sending a NOTIFY message.

26.9.4.2 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.

26.9.5 GFT-Control procedures for notifications

26.9.5.1 Actions at a node which generates notifications

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

26.9.5.2 Actions at a receiving node

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

26.9.5.2.1 Actions at a Transit node

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

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

26.9.5.2.2 Actions at a Receiving End node

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

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

> *NOTE. Further (implementation specific) actions of a node 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 specification.*

26.10 Interworking with (narrowband) QSIG

Two means exist for interworking with the narrowband private network. In the first the generic functional protocol is fully terminated. In the second, a generic interworking function is provided.

An interworking node shall provide the procedures of 26.10.1 for full termination of the PNNI and QSIG protocols. An interworking node may also provide the generic interworking procedures of 26.10.2 on a case by case basis.

When an interworking node receives generic functional data from a PNNI link and is able to support the optional procedures of 26.10.2, the decision to pass the generic functional data on to the QSIG link unchanged in accordance with 26.10.2 shall be based on the contents of the NFE (if any) as shown in table 26-8.

Likewise, when an interworking node receives generic functional data from a QSIG link and is able to support the optional procedures of 26.10.2, the decision to pass the generic functional data on to the PNNI link unchanged in accordance with 26.10.2 shall be based on the contents of the NFE (if any) as shown in table 26-8.

Table 26-8: Effect of NFE on handling of generic functional data at a PNNI to QSIG interworking node

NFE contents	Action
No NFE	Terminate in accordance with 26.10.1
NFE with destinationEntity value endNode	Optionally provide generic interworking; otherwise terminate in accordance with 26.10.1
NFE with destinationEntity value anyTypeOfNode and no destinationEntityAddress	Optionally provide generic interworking; otherwise terminate in accordance with 26.10.1
NFE with destinationEntity value anyTypeOfNode and a destinationEntityAddress	Terminate in accordance with 26.10.1 if the address matches the interworking node'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 unrecognized. 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 node, full termination in accordance with 26.10.1 shall be employed.

26.10.1 Full termination of generic functional protocol

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

26.10.2 Generic interworking function

26.10.2.1 Architecture

Figure 26-4 shows the protocol architecture of this interworking mechanism.

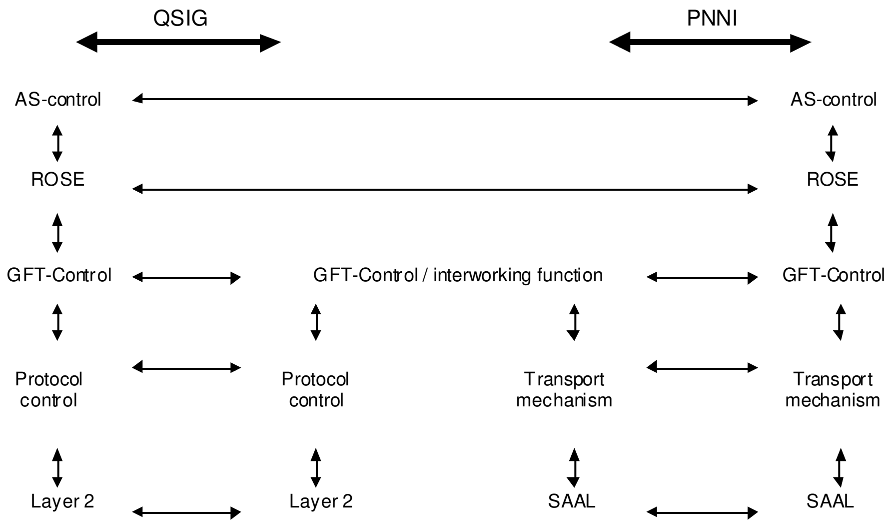


Figure 26-4: Generic interworking

For this form of interworking to take place, the supplementary service procedures or other functionality for both QSIG and PNNI 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 subsections.

26.10.2.2 Bearer-related transport mechanism

All mapping is performed as specified in section 6 with the addition that the Facility information element is included in all mapped messages.

In addition, for mapping PNNI to QSIG the mappings shown in table 26-9 shall apply.

Table 26-9: PNNI to QSIG mapping

PNNI message		QSIG message
FACILITY	----->	FACILITY

In addition, for mapping QSIG to PNNI, the mappings shown in table 26-10 shall apply.

Table 26-10: QSIG to PNNI mapping

QSIG message		PNNI message
FACILITY	----->	FACILITY

The PNNI 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 PNNI 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 6.5.6 apply.

26.10.2.3 Connection-oriented bearer independent mechanism

For mapping PNNI to QSIG , the mappings shown in table 26-11 shall apply.

Table 26-11: PNNI to QSIG mapping

PNNI 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 PNNI entity by the interworking function.

NOTE 2. A RELEASE COMPLETE message is also returned to the PNNI entity by the interworking function.

NOTE 3. This mapping only occurs if the PNNI RELEASE COMPLETE message is the first clearing message.

NOTE 4. It is not expected that this message would occur in a PNNI to QSIG interworking scenario.

For mapping QSIG to PNNI, the mappings shown in table 26-12 shall apply.

Table 26-12: QSIG to PNNI mapping

QSIG message		PNNI 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 tables 26-11 and 26-12 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 PNNI 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 PNNI 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 PNNI 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 6.5.6 shall apply.

26.10.2.4 Connectionless bearer independent mechanism

For mapping PNNI to QSIG, the mappings shown in table 26-13 shall apply.

Table 26-13: PNNI to QSIG mapping

PNNI message		QSIG message
FACILITY	----->	FACILITY

For mapping QSIG to PNNI, the mappings shown in table 26-14 shall apply.

Table 26-14: QSIG to PNNI mapping

QSIG message		PNNI message
FACILITY	----->	FACILITY

For the mappings shown in tables 26-13 and 26-14 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 PNNI protocol are discarded:

- Notification indicator information element;
- Designated transit list;
- Connection scope selection.

The PNNI 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 PNNI 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 6.5.6 apply.

26.11 Parameter values

26.11.1 Connection-oriented bearer-independent transport

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

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

26.12 Manufacturer Specific Information (MSI)

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

Manufacturer specific information may exist in the PNNI network 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. Any manufacturer specific information generated by a node conforming to this specification shall be encoded in conformance with the contents of this subsection.

26.12.1 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 subsections of this Annex, 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 Appendix N.

26.12.2 Manufacturer specific additions to standardised operations and errors

As an alternative to the definition of a manufacturer specific operation or error, a manufacturer may wish to use an enhanced 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 specifications 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 27.8 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 or 28.8 of Annex N 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 node 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 specification or additional basic call capability, it shall act on an element of type Extension only if it recognizes the value in the element of type OBJECT IDENTIFIER, (see 27.8 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 or the EXTENSION object class specified in 28.8 of Annex N 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 node shall consider each element of type Extension separately - that is, only those elements of type Extension containing an unrecognized value in the element of type OBJECT IDENTIFIER shall be discarded.

A manufacturer specific extension may be defined using the EXTENSION macro specified in 27.8 of Annex M using ASN.1 as specified in ITU-T Recommendation X.208 or the EXTENSION object class specified in 28.8 of Annex N using ASN.1 as specified in ITU-T Recommendation X.680.

An example of the use of the Extension type is shown in Appendix N.

26.12.3 Manufacturer specific notifications

Manufacturer specific notifications may occur in the PNNI network as part of manufacturer specific Supplementary services or as additions to standardised supplementary services. If provided, they shall be encoded and transported across the PNNI network in accordance with the rules for standardised notification (see 26.9.1.1, 26.8.1 and 26.8.2.3).

Manufacturer specific notification shall be conveyed using ASN.1 type NotificationDataStructure in octet 5.1 of the Notification indicator information element, as specified in 26.8.2.2.3.

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 26.8.2.2.3).

26.13 Compatibility with nodes that do not support this Annex

Messages and information elements introduced by this Annex will not be recognized by PNNI nodes that do not support GSS. In addition, a node that supports GSS is not required to support all three transport mechanisms (bearer-related, CO-BI and CL-BI). Therefore messages and information elements introduced for a specific transport mechanism will not be recognized by nodes that do not support that transport mechanism.

26.13.1 Bearer-related transport mechanism

Nodes that do not support bearer-related transport will not recognize the FACILITY message and Facility information element.

Subsection 25.9.2.1.1 makes provision for GFT-Control at a source node to specify the instruction indicator to be used in the Facility information element when transporting generic functional data. Subsection 25.9.1.1.1 requires the inclusion of this instruction indicator in the transmitted Facility information element and a corresponding instruction indicator in the FACILITY message, if this is the message selected for transport. Subsection 25.9.2.1.2.2 makes provision for the passing on of the instruction indicator at a transit node.

For local information exchange (no NFE present), the instruction indicator pass along request bit shall be set to “No pass along request”. It is recommended that the action indicator be set to “discard information element and proceed”, “discard information element, proceed and report status” or “clear call”, depending on the nature of the generic functional data being transported.

> *NOTE. The value “discard information element, proceed and report status” can lead to receipt of a STATUS message if the adjacent node does not support the GSS bearer-related transport mechanism.*
>
> *NOTE. The values “discard message and ignore” and “discard message and report status” should not be used, since GFT-Control is not necessarily aware of other information in the message.*

For non-local information exchange (NFE present), the instruction indicator pass along request bit shall be set to “Pass along request”. It is recommended that the action indicator be set to either “discard information element and proceed” or “clear call”, depending on the nature of the generic functional data being transported.

> *NOTE. The value “discard information element, proceed and report status” should not be used, since a STATUS message generated by a node that is not an adjacent node will not be passed back to the source node.*
>
> *NOTE. The values “discard message and ignore” and “discard message and report status” should not be used, since GFT-Control is not necessarily aware of other information in the message.*

26.13.2 CO-BI transport mechanism

Subsections 25.9.2.2.1.1 and 25.9.2.2.2.1 require observance of the nodal information flag “CO-BI transport supported” when selecting a route for a CO-BI SETUP message. This ensures that every node passed through supports the CO-BI SETUP message, the FACILITY message and the Facility information element, and therefore compatibility problems are avoided.

26.13.3 CL-BI transport mechanism

Subsections 25.9.2.3.1 and 25.9.2.3.2 require observance of the nodal information flag “CL-BI transport supported” when selecting a route for a FACILITY message with dummy call reference for CL-BI transport. This ensures that every node passed through supports the FACILITY message with dummy call reference and the Facility information element, and therefore compatibility problems are avoided.

26.13.4 Routing protocol

The two new nodal information flags introduced in changes to 5.8.1.2.3 (see Part 2 of this Addendum) mean that a node that does not support this Annex will ignore these flags in received PTSEs and set these flags to zero when originating PTSEs. In accordance with the last sentence of 5.8.3.2, a node passing on received PTSEs will not alter these flags.

27. Annex M: Formal definition of data types using ITU-T Recommendation X.208

This Annex is applicable only if optional Annex L is supported.

This Annex provides the ASN.1 modules referred to in Annex L.

27.1 ROSE APDU types

Table 27-1 shows the formal definition of the ROSE APDU data types used in the functional protocol.

Table 27-1: ROSE APDU types

```
Remote-Operations-Apdus
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x) 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
```

27.2 Definition of embedded PNNI information elements

Table 27-2 contains the ASN.1 definition of a general applicable type used to include PNNI information elements in ASN.1 definitions.

The PNNI information elements to be used shall be indicated as comment at the point where the type PnniInformationElement is used.

Table 27-2: Definition of embedded PNNI information elements

```
pnni-generic-parameter-definition
          {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures (x)
                    pnni-generic-parameters(2) }

DEFINITIONS EXPLICIT TAGS ::=
BEGIN
EXPORTS   PnniInformationElement;

PnniInformationElement ::= [APPLICATION 0]   IMPLICIT   OCTET STRING

END    -- of Pnni-generic-parameter-definition
```

27.3 Network facility extension

Table 27-3 contains the ASN.1 definition of type NetworkFacilityExtension.

Table 27-3: Network Facility Extension Coding

```
Network-Facility-Extension
          {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures ( x)
                    network-facility-extension( 3) }

DEFINITIONS ::=
BEGIN
EXPORTS   NetworkFacilityExtension;
IMPORTS   PartyNumber FROM Addressing-Data-Elements
          {         iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
                                        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
                               {       endNode ( 0),
                                       anyTypeOfNode ( 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
```

27.4 NOTIFICATION macro and notification for conveying embedded PNNI information elements

Table 27-4 defines the ASN.1 NOTIFICATION macro used for defining notifications that can be carried in the Notification indicator as defined in 26.8.2.2.3. It also defines the notification value pnniIeNotification, the use of which is described in 26.8.2.2.3.

Table 27-4: Notification macro definition

```
Notification-macro
          {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures ( x) notification-macro (4)
}

DEFINITIONS ::=
BEGIN
EXPORTS          NOTIFICATION, pnniIeNotification;
IMPORTS          PnniInformationElement     FROM    Pnni-generic-parameter-definition
                 {          iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
pnni-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 PNNI network

pnniIeNotification          NOTIFICATION
                                      ARGUMENT        PnniInformationElement
                                      ::= 2501

END     -- of Notification-macro
```

27.5 Addressing information definition

Table 27-5 contains the definition of ASN.1 types for encoding addressing information

Table 27-5: Addressing data elements

```
Addressing-Data-Elements
          {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures ( x)
                              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 Recommendation 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),
                              privateNetworkSpecificNumber (3),
                              localNumber (4),
                              abbreviatedNumber (6)                   }

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

PartySubaddress               ::=         CHOICE{
                              userSpecifiedSubaddress     UserSpecifiedSubaddress,
                                          -- not recommended
                              nSAPSubaddress              NSAPSubaddress                        }
                                          -- according to Recommendation 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
```

27.6 Interpretation APDU

Table 27-6 contains the ASN.1 definition of type Interpretation APDU.

Table 27-6: Interpretation APDU

```
Interpretation-Apdu
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x) interpretation-apdu (6) }

DEFINITIONS ::=
BEGIN
EXPORTS                    InterpretationApdu;

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

     }
END                 -- of Interpretation-Apdu
```

27.7 Notification Data Structure

Table 27-7 contains the ASN.1 definition of type NotificationDataStructure.

Table 27-7: ASN.1 encoded Notification Data Structure

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

```
DEFINITIONS ::=
BEGIN
EXPORTS                 NotificationDataStructure;

IMPORTS         NOTIFICATION    FROM    Notification-Macro
        { iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
                                        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
```

27.8 EXTENSION macro and Extension data type

Table 27-8 contains the ASN.1 definition of type Extension and macro EXTENSION.

Table 27-8: Manufacturer specific extension mechanism

```
Manufacturer-specific-service-extension-definition
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x) 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
```

28. Annex N: Formal definition of data types using ITU-T Recommendation X.680

This Annex is applicable only if optional Annex L is supported.

This Annex provides the ASN.1 modules referred to in Annex L.

28.1 ROSE APDU types

Table 28-1 shows the formal definition of the APDU data types used in the functional protocol.

Table 28-1: APDU types

```
Revised-Remote-Operations-Apdus
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x)
                              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-unrecognizedPDU)
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-unrecognizedLinkedId)
                                                   (CONSTRAINED BY {-- which has one or more linked
operations--}
                                              ! RejectProblem : invoke-linkedResponseUnexpected)
                                                         OPTIONAL,
                        opcode   OPERATION.&operationCode
                                                   ({Operations}
                                                   ! RejectProblem : invoke-unrecognizedOperation),
                        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-unrecognizedInvocation)
                                        (CONSTRAINED BY {-- which returns a result --}
                                        ! RejectProblem : returnResult-resultResponseUnexpected),
                        result  SEQUENCE
                {
                                opcode  OPERATION.&operationCode
                                        (({Operations})(CONSTRAINED BY {-- identified by invokeId --}
                                        ! RejectProblem : returnResult-unrecognizedInvocation)),
                                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-unrecognizedInvocation)
                                        (CONSTRAINED BY {-- which returns an error --}
                                        ! RejectProblem : returnError-errorResponseUnexpected),
                        errcode ERROR.&errorCode
                                                ({Errors}
                                                ! RejectProblem : returnError-unrecognizedError)
                                                (CONSTRAINED BY
                                                {--must be in the &Errors field of the associated operation --}
                                                        ! 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
                {
                        unrecognizedComponent (0),
                        mistypedComponent (1),
                        badlyStructuredComponent (2)
                }
InvokeProblem           ::=     INTEGER
                {
                        duplicateInvocation (0),
                        unrecognizedOperation (1),
                        mistypedArgument (2),
                        resourceLimitation (3),
                        releaseInProgress (4),
                        unrecognizedLinkedId (5),
                        linkedResponseUnexpected (6),
                        unexpectedLinkedOperation (7),
                }
ReturnResultProblem     ::=     INTEGER
                {
                        unrecognizedInvocation (0),
                        resultResponseUnexpected (1),
                        mistypedResult (2)
                }
ReturnErrorProblem       ::=    INTEGER
                {
                        unrecognizedInvocation (0),
                        errorResponseUnexpected (1),
                        unrecognizedError (2),
                        unexpectedError (3),
                        mistypedParameter (4)
                }
RejectProblem            ::=            INTEGER
                {
                        general-unrecognizedPDU (0),
                        general-mistypedPDU (1),
                        general-badlyStructuredPDU (2),
                        invoke-duplicateInvocation (10),
                        invoke-unrecognizedOperation (11),
                        invoke-mistypedArgument (12),
                        invoke-resourceLimitation (13),
                        invoke-releaseInProgress (14),
                        invoke-unrecognizedLinkedId (15),
                        invoke-linkedResponseUnexpected (16),
                        invoke-unexpectedLinkedOperation (17),
                        returnResult-unrecognizedInvocation (20),
                        returnResult-resultResponseUnexpected (21),
                        returnResult-mistypedResult (22),
                        returnError-unrecognizedInvocation (30),
                        returnError-errorResponseUnexpected (31),
                        returnError-unrecognizedError (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
```

28.2 Definition of embedded PNNI information elements

Table 28-2 contains the ASN.1 definition of a general applicable type used to include PNNI information elements in ASN.1 definitions.

The PNNI information elements to be used shall be indicated as comment at the point where the type PnniInformationElement is used.

Table 28-2: Definition of embedded PNNI information elements

```
Pnni-generic-parameter-definition
        {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures (x)
                        pnni-generic-parameters(12) }

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

28.3 Network facility extension

Table 28-3 contains the ASN.1 definition of type NetworkFacilityExtension.

Table 28-3: Network Facility Extension Coding

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

DEFINITIONS ::=
BEGIN
EXPORTS         NetworkFacilityExtension;
IMPORTS         PartyNumber FROM Addressing-Data-Elements
                {       iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
                                                                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
```

```
                                {       endNode ( 0),
                                        anyTypeOfNode ( 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
```

28.4 NOTIFICATION object class and notification for conveying embedded PNNI information elements

Table 28-4 defines the ASN.1 NOTIFICATION object class used for defining notifications that can be carried in the Notification indicator as defined in 26.8.2.2.3. It also defines the notification pnniIeNotification, the use of which is described in 26.8.2.2.3.

Table 28-4: Notification object class definition

```
Notification-object-class
        {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures ( x)
                        notification-object-class (14) }

DEFINITIONS ::=
BEGIN
EXPORTS         NOTIFICATION, pnniIeNotification;
IMPORTS         PnniInformationElement     FROM    Pnni-generic-parameter-definition
                {       iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
                                        pnni-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 PNNI network

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

28.5 Addressing information definition

Table 28-5 contains the definition of ASN.1 types for encoding addressing information

Table 28-5: Addressing information definitions

```
Addressing-Data-Elements
        {iso(1) identified-organization(3) icd-ecma(0012) standard (0) pnni-generic-procedures ( x)
                        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 Recommendation 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),
                        privateNetworkSpecificNumber (3),
                        localNumber (4),
                        abbreviatedNumber (6)           }
```

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

                PartySubaddress         ::=     CHOICE{
                                        userSpecifiedSubaddress  UserSpecifiedSubaddress,
                                                -- not recommended
                                        nSAPSubaddress           NSAPSubaddress            }
                                                -- according to Recommendation 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
```

28.6 Interpretation APDU

Table 28-6 contains the ASN.1 definition of type Interpretation APDU

Table 28-6: Interpretation APDU

```
Interpretation-Apdu
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x) interpretation-apdu (16) }

DEFINITIONS ::=
BEGIN
EXPORTS                 InterpretationApdu;

InterpretationApdu          ::=         [11] IMPLICIT ENUMERATED
        {       discardAnyUnrecognizedInvokePdu ( 0),
                clearCallIfAnyInvokePduNotRecognized ( 1),
```

```
                    -- This value also applies to CO-BI connections,
          rejectAnyUnrecognizedInvokePdu ( 2)
                    -- This coding is implied by the absence of an interpretation APDU

     }
END          -- of Interpretation-Apdu
```

28.7 Notification Data Structure

Table 28-7 contains the ASN.1 definition of type NotificationDataStructure.

Table 28-7: ASN.1 encoded Notification Data Structure

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

DEFINITIONS ::=
BEGIN
EXPORTS                  NotificationDataStructure{};

IMPORTS          NOTIFICATION   FROM    Notification-object-class
     {iso ( 1) identified-organization(3) icd-ecma(0012) standard ( 0) pnni-generic-procedures ( x)
                              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
```

28.8 EXTENSION object class and Extension data type

Table 28-8 contains the ASN.1 definition of the EXTENSION object class and type Extension.

Table 28-8: Manufacturer specific extension mechanism

```
Manufacturer-specific-service-extension-definition
 { iso(1) identified-organization(3) icd-ecma(0012) standard(0) pnni-generic-procedures (x) 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
```

29. Annex O: Conformance Requirements for the Generic Functional Protocol for the Support of Supplementary Services

This Annex is applicable only if optional Annex L is supported.

Conformance to the generic functional protocol for the support of supplementary services, as specified in Annexes L, M and N is optional. If implemented, those functions designated as mandatory in table 29-1 shall be implemented and those functions designated as optional in table 29-1 may be implemented.

As a general principle, for a given transport mechanism (items 1, 2 and 3 in table 29-1), by implementing just those capabilities designated as mandatory, an implementation that does not support individual supplementary services should be able to operate satisfactorily in a network with other implementations that do support one or more supplementary services. This minimum support includes the ability to pass on received information for which the node is not the destination and handle in a clean way (e.g., by rejection) received information for which the node is the destination.

Some of the capabilities that are designated as optional may become mandatory as a result of supporting certain supplementary services.

Table 29-1: Mandatory and optional capabilities relating to transport mechanisms

Capability no.	Capability	Status
1	Bearer-related transport of supplementary service information	O_1 (NOTE 1)
2	Connectionless bearer-independent (CL-BI) transport of supplementary service information	O_1 (NOTE 1)
3	Connection-oriented bearer-independent (CO-BI) transport of supplementary service information	O_1 (NOTE 1)
4	Minimum support of ROSE APDUs and the Interpretation APDU (NOTE 2)	M
5	ROSE as an invoker or performer of an operation	O (NOTE 1)
6	Generation of ASN.1-encoded notifications	O (NOTE 1)
7	Interworking with narrowband QSIG	O

O_1: Optional, but at least one of these three capabilities shall be implemented.

M: Mandatory

O: Optional

NOTE 1. Each of these optional items is required to be supported only if required by a particular supplementary service that is supported.

NOTE 2. This involves correct behaviour on receipt of an unrecognized APDU (not a valid ROSE APDU), a badly structured APDU, an invoke APDU containing an unrecognized (unsupported) operation value or an unexpected return result or return error APDU. In the case of an invoke APDU containing an unrecognized (unsupported) operation value, this also involves taking into account any received Interpretation APDU.

30. Annex P: Protocol Implementation Conformance Statement (PICS) proforma for Generic Support for Supplementary Services[2]

This Annex is applicable only if optional Annex L is supported.

This Annex provides the PICS proforma for Generic Support for Supplementary Services. It is an addition to the PNNI PICS proforma, as specified in ATM Forum PNNI v1.0 Errata and PICS (af-pnni-81.00, March 1997).

The supplier of a protocol implementation which is claimed to conform to PNNI version 1.0 and also to Generic Support for Supplementary Services, as specified in Annex L, is required to complete a copy of the PICS proforma provided in subsections 30.1 onwards and attach it to the completed copy of the PICS proforma for PNNI 1.0.

30.1 Generic support for supplementary services

Table 30-1: Provision of generic support for supplementary services

Item	Does the implementation…	Conditions for status	Status	Reference	Support
GSS 1	Provide generic support for supplementary services?		O		[ ]Yes [ ]No
Comments:					

---

[2] **Copyright release for PICS:** This PICS proforma may be freely reproduced, so that it may be used for its intended purpose.

30.2 Roles

Table 30-2: Type of implementation

Prerequisite: GSS 1

Tem	Major role: Does the implementation...	Conditions for status	Status	Reference	Support
	Type of implementation				
R 1.1	Support transit node?		O.1		[ ]Yes [ ]No
R 1.2	Support originating node?		O.1		[ ]Yes [ ]No
R 1.3	Support terminating node?		O.1		[ ]Yes [ ]No
R 1.4	Support incoming gateway node?		O.1		[ ]Yes [ ]No
R 1.5	Support outgoing gateway node?		O.1		[ ]Yes [ ]No
O.1	Support of at least one of these options is required.				
Comments:					

30.3 Major capabilities

Table 30-3: Major capabilities

Prerequisite: GSS 1

Item	Major capability: Does the implementation...	Conditions for status	Status	Reference	Support
Transport mechanisms					
MC 2.1	Support bearer related transport mechanism?		O.3	26.9.1.1	[ ]Yes [ ]No
MC 2.2	Support bearer independent transport mechanism?		O.3	26.9.1.2	[ ]Yes [ ]No
MC 2.3	Support (bearer independent) connection-oriented transport mechanism?	MC 2.2 NOT MC 2.2	O.4 N/A	26.9.1.3	[ ]Yes [ ]No [ ]N/A
MC 2.4	Support (bearer independent) connectionless transport mechanism?	MC 2.2 NOT MC 2.2	O.4 N/A	26.9.1.4	[ ]Yes [ ]No [ ]N/A
Notifications					
MC 3	Support transport of notifications?		M	26.9.4	[ ]Yes [ ]No
MC 3.1	Support transport of bearer-related notifications?	MC 2.1 NOT MC 2.1	M N/A	26.9.4	[ ]Yes [ ]No [ ] N/A
MC 3.2	Support transport of bearer-independent notifications?	MC 2.2 NOT MC 2.2	M N/A	26.9.4	[ ]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	26.9.4	[ ]Yes [ ]No [ ] N/A
MC 3.4	Support transport of bearer-independent connectionless notifications?	MC 2.4 NOT MC 2.4	M N/A	26.9.4	[ ]Yes [ ]No [ ] N/A
GFT-control					
MC 4	Provide GFT-control as a source node?		M	26.9.2.1.1	[ ]Yes [ ]No
MC 5	Provide GFT-control as a receiving node?		M	26.9.2.1.2	[ ]Yes [ ]No
MC 6	act as a destination node?		M	26.9.2.1.3	[ ]Yes [ ]No
MC 7	Interwork with (narrowband) QSIG?	R 1.4 OR R 1.5 NOT (R 1.4 OR R 1.5)	O N/A	26.10	[ ]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	26.10.1	[ ]Yes [ ]No
MC 7.2	Provide interworking by generic interworking function?	MC 7 NOT MC 7	O N/A	26.10.2	[ ]Yes [ ]No
MC 8	Support transport of manufacturer specific operations and errors?		M	26.12.1	[ ]Yes [ ]No
MC 9	Support transport of manufacturer specific additions to standardised operations and errors?		M	26.12.2	[ ]Yes [ ]No
MC 10	Support transport of manufacturer specific notification?		M	26.12.3	[ ]Yes [ ]No
MC 11	Behave as the invoker of remote operations?		O	26.9.3	[ ]Yes [ ] No NOTE 1
MC 12	Behave as the performer of remote operations?		O	26.9.3	[ ]Yes [ ] No NOTE 1
MC 13	Provide minimal support for ROSE when unable to behave as an invoker or performer of operations?	NOT (MC 11 OR MC12) MC 11 OR MC12	M N/A	26.9.3	[ ]Yes (NOTE 2) N/A

O.3	Support of at least one of these options is required.
O.4	Support of at least one of these options is required

NOTE 1. If the answer to this question is “Yes”, the PICS proforma for ROSE (X.249) also applies.

NOTE 2. Minimal support means that an implementation that can behave neither as an invoker of operations nor as a performer of operations can still find itself in a position, in accordance with the rules of GFT-Control, of being required to act as the destination node for APDUs. Such APDUs need to be handled in accordance with ROSE exceptional procedures and may involve the transmission of a Reject APDU in response..

Comments:

30.4 Subsidiary capabilities

Table 30-4: Subsidiary capabilities

Prerequisite: GSS 1

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	26.8.2.2.3	[ ]Yes [ ]No
SC 3.2	Support the transport of ASN.1 encoded notification information?		M	26.8.2.2.3, 27.7 and 28.7	[ ]Yes [ ]No

Comments:

30.5 Protocol data units

Table 30-5: Messages received

Prerequisite: GSS 1

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	26.8.1.3.2	[ ]Yes [ ]No [ ]N/A
MR 2	FACILITY?		M	26.8.1.1.1, 26.8.1.2.1, 26.8.1.3.4	[ ]Yes [ ]No
NOTE. These messages are additional to those required for support of basic call/connection.					
Comments:					

Table 30-6: Messages transmitted

Prerequisite: GSS 1

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	26.8.1.3.2	[ ]Yes [ ]No [ ]N/A
MT 1	FACILITY?		M	26.8.1.1.1, 26.8.1.2.1, 26.8.1.3.4	[ ]Yes [ ]No [ ]N/A
Comments:					

30.6 Protocol data unit parameters

30.6.1 Bearer-related transport mechanism

30.6.1.1 Protocol data unit parameters received

Table 30-7: ALERTING PDU parameters received

Prerequisite: R 1.1 or R 1.2 or R 1.4

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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 1.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-8: CONNECT PDU parameters received

Prerequisite: R 1.1 or R 1.2 or R 1.4

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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 2.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-9: FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 3.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.1 NOT MC 2.1	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 3.2	Facility?	MC 2.1 NOT MC 2.1	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 3.3	Notification indicator?	MC 2.1 AND MC 3.1 NOT (MC 2.1 AND MC 3.1)	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-10: 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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 4.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-11: 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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 5.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-12: 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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 6.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-13: SETUP PDU parameters received

Prerequisite: R 1.1 or R 1.3 or R 1.5

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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 7.2	Notification indicator?	MC 3.1 NOT MC 3.1	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

30.6.1.2 Protocol data unit parameters transmitted

Table 30-14: ALERTING PDU parameters transmitted

Prerequisite: R 1.1 or R 1.3 or R 1.5

Item	ALERTING PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 1.1	Facility?	MC 2.1 AND R 1.1 MC 2.1 AND NOT R 1.1 NOT MC 2.1	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 1.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-15: CONNECT PDU parameters transmitted

Prerequisite: R 1.1 or R 1.3 or R 1.5

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 2.1	Facility?	MC 2.1 AND R 1.1 MC 2.1 AND NOT R 1.1 NOT MC 2.1	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 2.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-16: FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 3.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.1 NOT MC 2.1	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 3.2	Facility?	MC 2.1 NOT MC 2.1	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 3.3	Notification indicator?	(MC 2.1 AND MC 3.1) AND R 1.1 (MC 2.1 AND MC 3.1) AND NOT R 1.1 NOT (MC2.1 AND MC 3.1)	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-17: 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 1.1 MC 2.1 AND NOT R 1.1 NOT MC 2.1	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 4.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-18: 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 1.1 MC 2.1 AND NOT R 1.1 NOT MC 2.1	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 5.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-19: 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	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 6.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

Table 30-20: SETUP PDU parameters transmitted

Prerequisite: R 1.1 or R 1.2 or R 1.4

Item	SETUP PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 7.1	Facility?	MC 2.1 AND R 1.1 MC 2.1 AND NOT R 1.1 NOT MC 2.1	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 7.2	Notification indicator?	MC 3.1 AND R 1.1 MC 3.1 AND NOT R 1.1 NOT MC 3.1	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
NOTE. These parameters are additional to those required for support of basic call/connection.					
Comments:					

30.6.2 Connection-oriented bearer-independent transport mechanism

30.6.2.1 Protocol data unit parameters received

Table 30-21: CALL PROCEEDING PDU parameters received

Prerequisite: R 1.1 or R 1.2 or R 1.4

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 8.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-22: CO-BI SETUP PDU parameters received

Prerequisite: R 1.1 or R 1.3 or R 1.5

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 9.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 9.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 9.3	Called party number?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.15	[ ]Yes [ ]No [ ]N/A
IER 9.4	Calling party number?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.17	[ ]Yes [ ]No [ ]N/A
IER 9.5	Designated transit list?	MC 2.3 NOT MC 2.3	M N/A	6.4.6.4	[ ]Yes [ ]No [ ]N/A
IER 9.6	Notification indicator?	MC 2.3 AND MC 3.3 NOT (MC 2.3 AND MC 3.3)	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-23: CONNECT PDU parameters received

Prerequisite: R 1.1 or R 1.2 or R 1.4

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 10.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 10.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 10.3	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-24: FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 11.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 11.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 11.3	Notification indicator?	MC 2.3 AND MC 3.3 NOT (MC 2.3 AND MC 3.3)	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-25: NOTIFY PDU parameters received

Item	NOTIFY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 12.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 12.2	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-26: RELEASE PDU parameters received

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 13.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 13.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 13.3	Crankback?	MC 2.3 NOT MC 2.3	M N/A	6.4.6.3	[ ]Yes [ ]No [ ]N/A
IER 13.4	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-27: 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, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 14.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 14.3	Crankback?	MC 2.3 NOT MC 2.3	M N/A	6.4.6.3	[ ]Yes [ ]No [ ]N/A
IER 14.4	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-28: STATUS PDU parameters received

Item	STATUS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 15.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 15.2	Call state?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.14	[ ]Yes [ ]No [ ]N/A
IER 15.5	Cause?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.19	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-29: 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, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
Comments:					

30.6.2.2 Protocol data unit parameters transmitted

Table 30-30: CALL PROCEEDING PDU parameters transmitted

Prerequisite: R 1.1 or R 1.3 or R 1.5

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 8.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-31: CO-BI SETUP PDU parameters transmitted

Prerequisite: R 1.1 or R 1.2 or R 1.4

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 9.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 9.2	Facility?	MC 2.3 AND R 1.1 MC 2.3 AND NOT R 1.1 NOT MC 2.3	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 9.3	Called party number?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.15	[ ]Yes [ ]No [ ]N/A
IET 9.4	Calling party number?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.17	[ ]Yes [ ]No [ ]N/A
IET 9.5	Designated transit list?	MC 2.3 NOT MC 2.3	M N/A	6.4.6.4	[ ]Yes [ ]No [ ]N/A
IET 9.6	Notification indicator?	(MC 2.3 AND MC 3.3) AND R 1.1 (MC 2.3 AND MC 3.3) AND NOT R 1.1 NOT (MC 2.3 AND MC 3.3)	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-32: CONNECT PDU parameters transmitted

Prerequisite: R 1.1 or R 1.3 or R 1.5

Item	CONNECT PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 10.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 10.2	Facility?	MC 2.3 AND R 1.1 MC 2.3 AND NOT R 1.1 NOT MC 2.3	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 10.3	Notification indicator?	MC 3.3 AND R 1.1 MC 3.3 AND NOT R 1.1 NOT MC 3.3	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-33: FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 11.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 11.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 11.3	Notification indicator?	(MC 2.3 AND MC 3.3) AND R 1.1 (MC 2.3 AND MC 3.3) AND NOT R 1.1 NOT (MC 2.3 AND MC 3.3)	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-34: NOTIFY PDU parameters transmitted

Item	NOTIFY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 12.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 12.2	Notification indicator?	MC 3.3 NOT MC 3.3	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-35: RELEASE PDU parameters transmitted

Item	RELEASE PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 13.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 13.2	Facility?	MC 2.3 AND R 1.1 MC 2.3 AND NOT R 1.1 NOT MC 2.3	M O N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
IET 13.3	Crankback?	MC 2.3 NOT MC 2.3	O N/A	6.4.6.3	[ ]Yes [ ]No [ ]N/A
IET 13.4	Notification indicator?	MC 3.3 AND R 1.1 MC 3.3 AND NOT R 1.1 NOT MC 3.3	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-36: 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, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 14.2	Facility?	MC 2.3 NOT MC 2.3	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 14.3	Crankback?	MC 2.3 NOT MC 2.3	O N/A	6.4.6.3	[ ]Yes [ ]No [ ]N/A
IET 14.4	Notification indicator?	MC 3.3 AND R 1.1 MC 3.3 AND NOT R 1.1 NOT MC 3.3	M O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-37: STATUS PDU parameters transmitted

Item	STATUS PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 15.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 15.2	Call state?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.14	[ ]Yes [ ]No [ ]N/A
IET 15.3	Cause?	MC 2.3 NOT MC 2.3	M N/A	6.4.5.19	[ ]Yes [ ]No [ ]N/A
Comments:					

Table 30-38: 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, Call reference, Message type and Message length?	MC 2.3 NOT MC 2.3	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
Comments:					

30.6.3 Connectionless bearer-independent transport mechanism

30.6.3.1 Protocol data unit parameters received

Table 30-39: FACILITY PDU parameters received

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IER 17.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.4 NOT MC 2.4	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IER 17.2	Facility?	MC 2.4 NOT MC 2.4	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IER 17.3	Called party number?	MC 2.4 NOT MC 2.4	M N/A	6.4.5.15	[ ]Yes [ ]No [ ]N/A
IER 17.4	Calling party number?	MC 2.4 NOT MC 2.4	M N/A	6.4.5.17	[ ]Yes [ ]No [ ]N/A
IER 17.5	Designated transit list?	MC 2.4 NOT MC 2.4	M N/A	6.4.6.4	[ ]Yes [ ]No [ ]N/A
IER 17.6	Notification indicator?	MC 3.4 NOT MC 3.4	M N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

30.6.3.2 Protocol data unit parameters transmitted

Table 30-40: FACILITY PDU parameters transmitted

Item	FACILITY PDU parameters: Does the implementation support the ...	Conditions for status	Status	Reference	Support
IET 17.1	Protocol discriminator, Call reference, Message type and Message length?	MC 2.4 NOT MC 2.4	M N/A	6.4.1, 6.4.2, 26.8.2.1 and 6.4.4	[ ]Yes [ ]No [ ]N/A
IET 17.2	Facility?	MC 2.4 NOT MC 2.4	M N/A	26.8.2.2.2	[ ]Yes [ ]No [ ]N/A
IET 17.3	Called party number?	MC 2.4 NOT MC 2.4	M N/A	6.4.5.15	[ ]Yes [ ]No [ ]N/A
IET 17.4	Calling party number?	MC 2.4 NOT MC 2.4	M N/A	6.4.5.17	[ ]Yes [ ]No [ ]N/A
IET 17.5	Designated transit list?	MC 2.4 NOT MC 2.4	M N/A	6.4.6.4	[ ]Yes [ ]No [ ]N/A
IET 17.6	Notification indicator?	MC 3.4 NOT MC 3.4	O N/A	26.8.2.2.3	[ ]Yes [ ]No [ ]N/A
Comments:					

30.7 Components of protocol data unit parameters

30.7.1 Facility information element

30.7.1.1 Components received

Table 30-41: Facility information element components received

Item	Facility information element: Does the implementation support reception of the ...	Conditions for status	Status	Reference	Support
IECR 1	Information element identifier, coding standard, information element instruction field and length of facility contents?		M	26.8.2.2.2	[ ]Yes
IECR 2	Protocol profile?		M	26.8.2.2.2	[ ]Yes
IECR 3	Network Facility Extension?	MC 2.1 OR MC 2.3 NOT (MC 2.1 OR MC 2.3)	M N/A	26.8.2.2.2.1, 27.3, 28.3	[ ]Yes [ ]N/A
IECR 4	Interpretation APDU?		M	26.8.2.2.2.2	[ ]Yes
IECR 5	ROSE APDUs?		M	26.8.2.2.2.3	[ ]Yes
Comments:					

30.7.1.2 Components transmitted

Table 30-42: Facility information element components transmitted

Item	Facility information element: Does the implementation support transmission of the ...	Conditions for status	Status	Reference	Support
IECR 1	Information element identifier, coding standard, information element instruction field and length of facility contents?		M	26.8.2.2.2	[ ]Yes
IECR 2	Protocol profile?		M	26.8.2.2.2	[ ]Yes
IECR 3	Network Facility Extension?	MC 2.1 OR MC 2.3 NOT (MC 2.1 OR MC 2.3)	M N/A	26.8.2.2.2.1, 27.3, 28.3	[ ]Yes [ ]N/A
IECR 4	Interpretation APDU?	R 1.1 NOT R 1.1	M O	26.8.2.2.2.2	[ ]Yes [ ]Yes [ ]No
IECR 5	ROSE APDUs?		M	26.8.2.2.2.3	[ ]Yes
Comments:					

31. Appendix J: Information flows

This Appendix is applicable only if optional Annex L is supported.

31.1 Connection-oriented bearer independent transport mechanism

31.1.1 Bearer independent establishment and data transfer

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

32. Appendix K: Instruction indicators

This Appendix is applicable only if optional Annex L is supported.

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

33. Appendix L: Formal definitions of remote operations notation using ITU-T Recommendation X.208

This Appendix is applicable only if optional Annex L is supported.

Table 33-1: Formal definition of remote operations data types
(extract of ITU-T Recommendation 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
```

34. Appendix M: Formal definitions of remote operations notation using ITU-T Recommendation X.680

This Appendix is applicable only if optional Annex L is supported.

Table 34-1: Formal definition of remote operations data types
(extract of ITU-T Recommendation 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
```

35. Appendix N: Examples of the use of Manufacturer Specific Information

This Appendix is applicable only if optional Annex L is supported.

35.1 Manufacturer Specific Object Identifier in Operation Values

As defined in 26.12.1, 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 organization" according to ISO/IEC 6523 and be assigned an organization identifier. The manufacturer should then use that organization identifier in an object identifier (as the root of the manufacturer specific service operation value) according to the structure defined by the issuing organization.

One example of regional Sponsoring and issuing organization 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 organization identifiers in object identifiers is as shown in table 35-1. nodes conforming to this specification can make use of an organization identifier issued by any "sponsoring and issuing organization" (e.g. ECMA or a national body).

Thus, according to table 35-1, the ECMA object identifier for a company with the assigned organization code '1999' (all organization codes issued by ECMA have 4 digits of which the first is always '1'), may be structured as shown in table 35-2. 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 35-3.

Table 35-1: Structure of ECMA Object Identifier

level 1: iso(1)
level 2: identified-organization (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: organization identifier assigned by ECMA
level 6: this level and others below it are used to suit the purpose of the organization assigned the value at level 5.

Table 35-2: ECMA Object identifier for hypothetical manufacturer specific service operation

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

          Hypothetical ManufacturerSpecificsupplementaryService ::=
               { iso( 1) identified-organization( 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 Recommendation X.209)
```

Table 35-3: Example of manufacturer specific operation

```
          Hypothetical-service-operation
          {    iso(1)    identified-organization(3)  icd-ecma(0012) -member-company(2)
                    hypothetical-manufacturer(1999)  hypothetical-service-offering(0)      }

DEFINITIONS ::=
BEGIN


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

          hypotheticalService   ::=   OPERATION
               {    ARGUMENT     HypotheticalArgument
                    RESULT       HypotheticalResult
                    ALWAYS RESPONDS FALSE
                    CODE global:{ iso ( 1) identified-organization ( 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
```

35.2 Manufacturer specific extensions to standardised supplementary services

An example of the use the element of type Extension (defined in 26.12.1) in a standardised supplementary services definition is given in table 35-4 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 specification 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{ExtensionSet},
                    [1]   IMPLICIT SEQUENCE OF
                                    Extension{ExtensionSet}
                    } 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{ExtensionSet},
               [1]   IMPLICIT         SEQUENCE OF
                                      Extension{ExtensionSet}
               } OPTIONAL
          }
```

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

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

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 35-3 above) should be encoded in the same manner as described in 26.12.1.

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 35-4: 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-Operations-Information-Objects
               {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)        }
          EXTENSION, Extension{} FROM Manufacturer-specific-service-extension-definition
          {    iso standard pnni-generic-procedures ( x) msi-definition ( 18)        };

          hypotheticalService       ::=      OPERATION
               {    ARGUMENT CHOICE
                         {  normal           NormalIntegerArgument,
                            extended         SEQUENCE
                            {        normal           NormalIntegerArgument,
                                     extension        CHOICE
                                     {        simple   [2]      IMPLICIT Extension{ExtensionSet},
                                              multiple [3]      IMPLICIT SEQUENCE OF Extension{ExtensionSet}
                                     }        OPTIONAL
                            }
                         }
                    RESULT          SEQUENCE
                         {  normal           ListOfNormalResultSequenceElements,
                            extension        CHOICE
                            {        simple   [2]      IMPLICIT Extension{ExtensionSet},
                                     multiple [3]      IMPLICIT SEQUENCE OF Extension{ExtensionSet}
                            }        OPTIONAL
                         }
                    ALWAYS RESPONDS  FALSE
                    CODE            global:{ iso standard hypothetical-standard ( 2222222)
                                             hypothetical-operation ( 10)          }
          }

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

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

ExtensionSet  EXTENSION  ::=  {...}

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

36. Appendix O: Remote operations protocol

This Appendix is applicable only if optional Annex L is supported.

The remote operations (RO) protocol is defined in ITU-T Recommendation 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 specification provide an encoding mechanism for the transport and use of this RO protocol for the provision of Supplementary services or additional basic call capabilities in a PNNI network.

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 0-1 models the view.

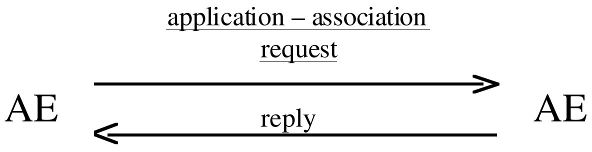


Figure 0-1: Remote operation model

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 specification).

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 0-2 models this concept.

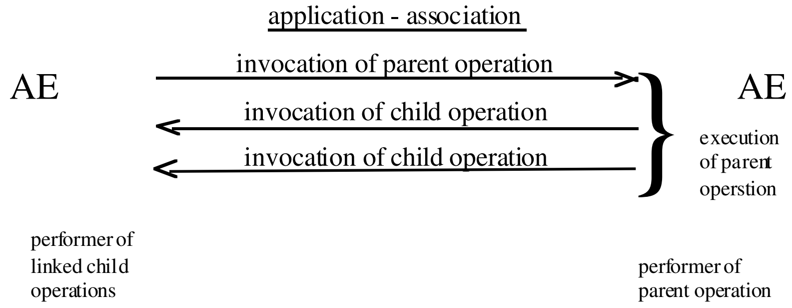


Figure 0-2: Linked operations

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 PNNI 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 specification assumes Application associations of Association Class 3.

37. Appendix P: Problem code definitions

This Appendix is applicable only if optional Annex L is supported.

Table 37-1: 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, subsection B.1 of this specification
–	mistypedAPDU	signifies that the structure of the APDU does not conform to 27.1 or 28.1
–	badlyStructuredAPDU	signifies that the structure of the APDUs does not conform to the standard notation and encoding, defined in ITU-T Recommendation X.208 and X.209, or ITU-T Recommendations X.680 and X.690 as appropriate
Invoke-problem:		
–	duplicateInvocation	signifies that the invoke-identifier parameter violates the assignment rules of ITU-T Recommendation 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 node 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.

PART 2: Changes to existing subsections and Annexes

Changes to section 5

The following text replaces the first paragraph of 5.8.1.2.3:

> Currently seven flags are defined: the “I am leader” flag, the Restricted Transit flag, the Nodal Representation flag, the Restricted Branching flag, the Non-transit for PGL Election flag, the CO-BI transport supported flag and the CL-BI transport supported flag. The rest of this field is reserved for use in future versions of PNNI.

The following paragraphs are added at the end of 5.8.1.2.3:

> The CO-BI transport supported flag is set to one by a node that supports the Connection-Oriented Bearer-Independent (CO-BI) transport mechanism as part of Generic Support for Supplementary Services. This flag is set to zero by any other node. Use of this flag is specified in Annex L.
>
> The CL-BI transport supported flag is set to one by a node that supports the Connectionless Bearer-Independent (CL-BI) transport mechanism as part of Generic Support for Supplementary Services. This flag is set to zero by any other node. Use of this flag is specified in Annex L.

The last column (bits 3..1) of table 5-36 (nodal flags) in 5.14.9.1.2 is replaced by the following:

bit 3	**bit 2**	**bit 1**
CO-BI transport supported flag	CL-BI transport supported flag	Reserved
Value 0: CO-BI transport not supported Value 1: CO-BI transport supported	Value 0: CL-BI transport not supported Value 1: CL-BI transport supported	

Changes to Annex G

Table 13-1 in Annex G is extended with the following entries:

No.	**Capabilities**	**Switching System**
44	B-QSIG interworking	O
45	Generic Support for Supplementary Services	O (Note 3)

NOTE 3. Sub-capabilities within Generic Support for Supplementary Services are detailed in Annex O

Changes to Annex H (MIB)

Object type PnniNodeEntry is extended by adding two new elements as indicated by underlining below:

```
PnniNodeEntry ::=
        SEQUENCE {
                pnniNodeIndex           PnniNodeIndex,
                pnniNodeLevel           PnniLevel,
```

```
                pnniNodeId              PnniNodeId,
                pnniNodeLowest          TruthValue,
                pnniNodeAdminStatus     INTEGER,
                pnniNodeOperStatus      INTEGER,
                pnniNodeDomainName      DisplayString,
                pnniNodeAtmAddress      PnniAtmAddr,
                pnniNodePeerGroupId     PnniPeerGroupId,
                pnniNodeRestrictedTransit       TruthValue,
                pnniNodeComplexRep      TruthValue,
                pnniNodeRestrictedBranching     TruthValue,
                pnniNodeDatabaseOverload        TruthValue,
                pnniNodePtses           Gauge32,
                pnniNodeRowStatus       RowStatus,
                pnniNodeCoBiTransportSupported  TruthValue,
                pnniNodeClBiTransportSupported  TruthValue
                }
```

The following new object types are added after object type pnniNodeRestrictedBranching:

```
pnniNodeCoBiTransportSupported OBJECT-TYPE
        SYNTAX        TruthValue
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
            "Specifies whether the node supports CO-BI transport as part
             of generic support for supplementary services (see Annex L).
             This attribute determines the setting of the CO-BI transport
             supported bit in the nodal information group originated by
             this node."
        REFERENCE
            "ATM Forum PNNI 1.0 Section 5.8.1.2.3"
        ::= { pnniNodeEntry 16 }

pnniNodeClBiTransportSupported OBJECT-TYPE
        SYNTAX        TruthValue
        MAX-ACCESS    read-create
        STATUS        current
```

```
        DESCRIPTION
            "Specifies whether the node supports CL-BI transport as part
             of generic support for supplementary services (see Annex L).
             This attribute determines the setting of the CL-BI transport
             supported bit in the nodal information group originated by
             this node."
        REFERENCE
            "ATM Forum PNNI 1.0 Section 5.8.1.2.3"
        ::= { pnniNodeEntry 17 }
```

The following new object group is added prior to the END statement:

```
pnniNodeGssOptionalGroup OBJECT-GROUP
        OBJECTS {
                pnniNodeCoBiTransportSupported,
                pnniNodeClBiTransportSupported
                }
        STATUS current
        DESCRIPTION
                "A collection of optional per-node PNNI objects used for
management of
                generic support for supplementary services."
        ::= { pnniMIBGroups 32}
```

– END OF DOCUMENT –