INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Q.932

Amendment 1
(06/2000)

SERIES Q: SWITCHING AND SIGNALLING

Digital subscriber Signalling System No. 1 – Network layer

Digital Subscriber Signalling System No. 1 – Generic procedures for the control of ISDN supplementary services

Amendment 1

ITU-T Recommendation Q.932 – Amendment 1

(Formerly CCITT Recommendation)

ITU-T Q-SERIES RECOMMENDATIONS

SWITCHING AND SIGNALLING

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4 AND No. 5	Q.120–Q.249
SPECIFICATIONS OF SIGNALLING SYSTEM No. 6	Q.250–Q.309
SPECIFICATIONS OF SIGNALLING SYSTEM R1	Q.310–Q.399
SPECIFICATIONS OF SIGNALLING SYSTEM R2	Q.400–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.799
Q3 INTERFACE	Q.800–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
General	Q.850–Q.919
Data link layer	Q.920–Q.929
Network layer	**Q.930–Q.939**
User-network management	Q.940–Q.949
Stage 3 description for supplementary services using DSS1	Q.950–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
BROADBAND ISDN	Q.2000–Q.2999

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation Q.932

Digital Subscriber Signalling System No. 1 – Generic procedures for the control of ISDN supplementary services

AMENDMENT 1

Summary

This Amendment 1 to ITU-T Q.932 (1998) is provided in order to support the Generic Addressing and Transport protocol as defined in ITU-T Q.860 (2000). Additionally, various ASN.1 module definitions according to ITU-T X.680/X.880 are also added and necessary updates and corrections for existing ASN.1 modules are provided.

Source

Amendment 1 to ITU-T Recommendation Q.932 was prepared by ITU-T Study Group 11 (1997-2000) and approved under the WTSC Resolution 1 procedure on 15 June 2000.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSC Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

Recommendation Q.932

Digital Subscriber Signalling System No. 1 – Generic procedures for the control of ISDN supplementary services

AMENDMENT 1

Text and tables to ITU-T Q.932 are amended as follows:

1) Clause 1

Change the heading of the existing clause 1 to "Scope".

2) Subclause 1.1

A new subclause "1.1 Definitions" *should be added as follows*:

For the purpose of the GAT protocol, the following definitions apply:

1.1.1 local signalling: a signalling procedure restricted to the exchange of application protocol data units between adjacent application entities.

1.1.2 general signalling: a signalling procedure for the exchange of application protocol data units between application entities that need not be adjacent.

3) Subclause 6.1.1 Introduction

Insert a 3rd paragraph as follows:

The support of the GAT protocol as defined in ITU-T Q.860 is a network and user option and its use is based on bilateral agreement between the network and user.

The extensions of the generic functional protocol required for the support of the GAT protocol are described in ITU-T Q.860 (generic part) and in this Recommendation (transport mechanisms and coding of the facility information element).

4) Subclause 6.4.3 Effect of reservation on channel selection for a new call

The following new Table 6-1 bis is the ASN.1 module according to ITU-T X.680/X.880 and sould be added after Table 6-1 as follows:

Table 6-1 *bis*/Q.932 – Explicit network controlled channel reservation (ASN.1 module according to ITU-T X.680/X.880)

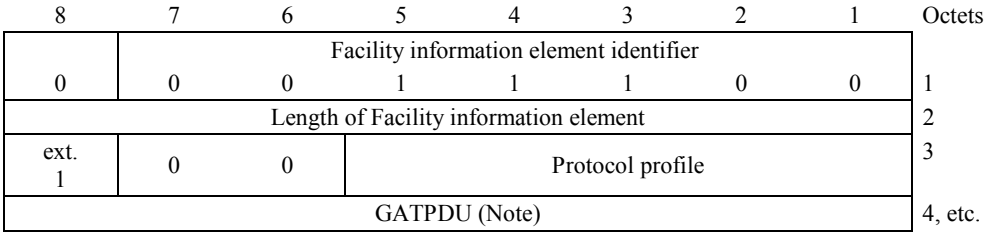
```
Explicit-Network-Controlled-Channel-Reservation
{itu-t recommendation q 932 explicit-network-controlled-channel-reservation(4)}

DEFINITIONS ::=

BEGIN

EXPORTS   explicitReservationCreationControl, explicitReservationManagement,
          explicitReservationCancel, maximumNumberOfReservationsReached,
          noExplicitReservationExistsOrInvalidReservationIndicator, unwantedReservationCreated,
          implicitReservationUsed, ReservationIndicator;

IMPORTS   OPERATION, ERROR
          FROM Remote-Operations-Information-Objects
          {joint-iso-itu-t remote-operations(4) informationObjects(5) version1(0)};
          notAvailable, notSubscribed
          FROM General-Errors {itu-t recommendation q 950 general-errors-list(1)};

explicitReservationCreationControl OPERATION ::=
          {
          ARGUMENT controlOption ENUMERATED {
                noReservationRequired (0),
                reservationRequiredWithReservationIndicator (1),
                reservationRequiredWithoutReservationIndicator (2)}
          RESULT ReservationIndicator -- optional
          ERRORS {
                maximumNumberOfReservationsReached|
                notAvailable|
                notSubscribed|
                unwantedReservationCreated}
          CODE          local:20
          }
explicitReservationManagement OPERATION ::=
          {
          ARGUMENT ReservationIndicator -- optional
          RESULT
          ERRORS {
                noExplicitReservationExistsOrInvalidReservationIndicator|
                notAvailable|
                notSubscribed|
                implicitReservationUsed}
          CODE          local:21
          }
explicitReservationCancel OPERATION ::=
          {
          ARGUMENT ReservationIndicator -- optional
          RESULT
          ERRORS {
                noExplicitReservationExistsOrInvalidReservationIndicator|
                notAvailable|
                notSubscribed}
          CODE          local:22
      }

maximumNumberOfReservationsReached                    ERROR ::= {CODE local:33}
noExplicitReservationExistsOrInvalidReservationIndicator  ERROR ::= {CODE local:34}
unwantedReservationCreated                            ERROR ::= {CODE local:35}
implicitReservationUsed                               ERROR ::= {CODE local:36}
ReservationIndicator ::=                              INTEGER (–128..127)

END -- of Explicit-Network-Controlled-Channel-Reservation
```

5) 8.2.3 Facility

a) *Insert the following text after the 4th paragraph:*

This subclause defines in addition the structure and the coding of the Facility information element, required for the support of the GAT protocol, as described in ITU-T Q.860. Specific procedures describing applications of the GAT protocol are outside the scope of this Recommendation.

b) *A new Figure 8-2 bis specifies the layout of the Facility information element for the GAT protocol and should be inserted after Figure 8-2 as follows:*



NOTE – When the protocol profile field in Octet 3 contains the value "GAT Protocol", a single GATPDU component (as defined in ITU-T Q.860) shall be included in octet group 4.

Figure 8-2 *bis*/Q.932 – Facility information element

c) *Replace Table 8-5 with the following:*

Table 8-5/Q.932 – Facility information element

Protocol Profile
Bits
5 4 3 2 1
1 0 0 0 1 Remote Operations Protocol (Note 1)
1 0 0 1 0 CMIP Protocol (see ITU-T Q.941 [11]) (Note 2)
1 0 0 1 1 ACSE Protocol (see ITU-T X.217 and X.227 [12]) (Note 3)
1 0 1 1 0 GAT Protocol (Note 5)
1 1 1 1 1 Networking extensions (Note 4)
All other values are reserved and their usage is the subject of other Recommendations.
NOTE 1 – When this codepoint is used, the NFE, NPP, and the Interpretation components shall be excluded. In addition, ITU-T defined local values apply for the components.
NOTE 2 – When this codepoint is used, the NFE, NPP, and the Interpretation components shall be excluded. See ITU-T Q.941 for the CMIP protocol.
NOTE 3 – When this codepoint is used, the NFE, NPP, and the Interpretation components shall be excluded. See ITU-T X.217 and X.227 for the ACSE protocol.
NOTE 4 – When this codepoint is used, the NFE, NPP, and the Interpretation components as defined in ISO/IEC 11582 can be included. In this case, only ISO/IEC defined local values apply for the components. For consistency with ISO/IEC 11582, a Facility information element with the protocol profile value "networking extensions" shall not be included in the SETUP ACKNOWLEDGE, CALL PROCEEDING, and CONNECT ACKNOWLEDGE messages as these messages are of local significance.
NOTE 5 – When this codepoint is used, a single GATPDU component as defined in ITU-T Q.GFT shall be included in octet group 4 of the Facility information element.

6) Subclause 8.2.3.1.1 Components (octets 4, etc.)

a) *Replace the definition of the return result component in Table 8-7 with the following:*

```
ReturnResultComponent ::=       SEQUENCE {
        Invokeid invokeIdentifierType,
        SEQUENCE {
        OperationValue OPERATION,
        Result ANY DEFINED BY
        OperationValue } OPTIONAL }
```

b) *A new Table 8-7 bis provides the coding of the Facility information element according to ITU-T X.680/X.880 and should be added after Table 8-7 as follows:*

Table 8-7 *bis*/Q.932 – Facility information element component coding

```
Facility-Information-Element-Components
        {itu-t recommendation q 932 facility-information-element-component(3)}
DEFINITIONS
IMPLICIT TAGS ::=
BEGIN

-- exports everything
IMPORTS OPERATION, ERROR FROM Remote-Operations-Information-Objects {joint-iso-itu-t
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
}
```

Table 8-7 *bis*/Q.932 – Facility information element component coding *(continued)*

```
(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
}
```

Table 8-7 *bis*/Q.932 – Facility information element component coding *(continued)*

```
(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)
}
```

Table 8-7 *bis*/Q.932 – Facility information element component coding *(concluded)*

```
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,
        absentNULL
}
noInvokeId InvokeId ::=   absent:NULL

NoInvokeId InvokeId ::=  {noInvokeId}

Errors {OPERATION:Operations} ERROR ::= {Operations.&Errors}

-- the Range of the Invoke-Id's must be restricted to –32768..32767
Components {OPERATION:Invokable, OPERATION:Returnable} ::=
{
gfpROS       ROS {GFPInvokeIDSet, {Invokable}, {Returnable}}
}
GFPInvokeIDSet INTEGER  ::= {-32768..32767}
END -- end of generic ROS PDU definitions
```

7) Subclause 8.2.8 Notification indicator

A new Table 8-14 bis provides the coding of the Notification indicator information element according to ITU-T X.680/X.880 and should be after Table 8-14 as follows:

Table 8-14 *bis*/Q.932 – Formal definition of the Notification indicator information element

```
Notification-Indicator-IE-Data-Structure {itu-t recommendation q 932 notification-data-structure (6)}

DEFINITIONS ::=
BEGIN

EXPORTS   NOTIFICATION;

NOTIFICATION ::=      CLASS
                      {
                            &ArgumentType                          OPTIONAL,
                            &operationCode       Code UNIQUE       OPTIONAL
                      }

WITH SYNTAX
                      {
                            [ARGUMENT            &ArgumentType]
                            [CODE                &operationCode]
                      }

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

NotificationDataStructure ::=     SEQUENCE
                {
                      opcode               NOTIFICATION.&operationCode
                      argument             NOTIFICATION.&ArgumentType
                }

END -- of Notification-Indicator-IE-Data-Structure
```

8) Subclause 10.1 Point-to-point Networked connection-oriented call independent component transport mechanism

Add the following text after the second paragraph:

The GAT protocol uses the Point-to-point Networked connection-oriented call independent component transport mechanism for the bearer-unrelated transport of GAT PDUs.

9) Clause 11

Insert a new clause 11 after subclause 10.2 as follows:

11 GFT Control and GAT Control

11.1 GFT Control

11.1.1 Transmission of generic functional data

When the ROSE entity or any other ASE requires to transmit generic functional data, this is indicated to the GFT-Control entity. Where general signalling is used, this indication comes via the GAT-Control entity (see 11.2). GFT-Control entity shall:

a) Determine from the information supplied by the ROSE entity or any other ASE the transport mechanism required.

   Where local signalling is required by the application, then the transport mechanisms that may be used are:

   – the point-to-point bearer-related transport mechanism defined in 6.3.1;
   – the broadcast bearer-related transport mechanism defined in 6.3.1;
   – the point-to-point transport mechanism (connection-oriented) defined in 6.3.2.1;
   – the broadcast transport mechanism (connection-oriented) defined in 6.3.2.2;
   – the broadcast transport mechanism (connectionless) defined in 6.3.2.3.

   Where general signalling is required by the application, then the transport mechanisms that may be used are:

   – the point-to-point bearer-related transport mechanism defined in 6.3.1;
   – the broadcast bearer-related transport mechanism defined in 6.3.1;
   – the point-to-point transport mechanism (connection-oriented) defined in 10.1.

b) Ensure that the required transport mechanism is in an appropriate state to transmit generic functional data.

c) Supply to the appropriate transport mechanism the generic functional data and protocol profile based on the type of ASE requesting transport of generic functional data. In particular:

   – for the support of local addressing within the constraints of this part of this Recommendation, the protocol profile shall be set to the value "ROSE";
   – for the support of general addressing, the protocol profile shall be set to the value "GAT Protocol".

d) For local signalling, indicate any address required for the multiple subscriber number or subaddressing supplementary service, if applicable.

e) For general signalling, indicate any address and identification information required for addressing and identification within the transport mechanism.

If the GFT-Control entity is unable to provide the transfer of generic functional data, it shall indicate this to the ROSE entity or any other appropriate ASE.

11.1.2 Receipt of generic functional data

When a transport mechanism receives a Facility information element, this is indicated to the GFT-Control entity.

The GFT-Control entity shall check the protocol profile, and if it is valid it shall indicate this data to the appropriate ASE. In particular, if the protocol profile is set to "ROSE", this shall be indicated to the ROSE entity. If the protocol profile is set to a reserved value, or to a value of protocol profile that is not supported, the generic functional data shall be discarded, and the procedures for unrecognized information element content handling as specified in 5.8.7.1 or 5.8.7.2/Q.931, as appropriate, shall be followed on the appropriate transport mechanism. The error handling rules in 5.8.7.1 and 5.8.7.2/Q.931 shall apply only for errors in the octets 1 through 3 of the Facility information element. Errors in subsequent octets shall be handled according to the appropriate ROSE protocol procedures.

11.2 GAT-Control

When general addressing is supported, the GAT-Control entity shall be provided, and the procedures for the GAT protocol specified in ITU-T Q.860 shall apply.

10) Annex C

a) *Replace the definition of the PartyNumber in Table C-1 with the following:*

```
PartyNumber          ::= CHOICE {
                             unknownPartyNumber [0] IMPLICIT NumberDigits,
                             -- the numbering plan is the default numbering plan of the
                             -- network. It is recommended that this value is used.
                             publicPartyNumber [1] IMPLICIT PublicPartyNumber,
                             -- the numbering plan is according to
                             -- ITU-T Recommendation 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, value reserved.

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

b) *A new Table C-1 bis provides the coding of the Addressing data elements according to ITU-T X.680/X.880 and should be added after Table C-1 as follows:*

Table C-1 *bis*/Q.932 – Addressing data elements
(ASN.1 module according to ITU-T X.680/X.880)

```
Addressing-Data-Elements {itu-t recommendation q 932 addressing-data-elements(7)}

DEFINITIONS EXPLICIT TAGS ::=

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 {
                                  presentationAllowedNumber [0] IMPLICIT NumberScreened,
                                  presentationRestricted [1] IMPLICIT NULL,
                                  numberNotAvailableDueToInterworking [2] IMPLICIT NULL,
                                  presentationRestrictedNumber [3] IMPLICIT NumberScreened}
```

Table C-1 *bis*/Q.932 – Addressing data elements
(ASN.1 module according to ITU-T X.680/X.880) *(continued)*

```
PresentedNumberUnscreened ::= CHOICE {
                                        presentationAllowedNumber [0] PartyNumber,
                                        presentationRestricted [1] IMPLICIT NULL,
                                        numberNotAvailableDueToInterworking [2] IMPLICIT NULL,
                                        presentationRestrictedNumber [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 value is used.
                                        publicPartyNumber [1] IMPLICIT PublicPartyNumber,
                                        -- the numbering plan is according to
                                        -- ITU-T 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, value reserved.
                                        …}

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

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

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

Table C-1 *bis*/Q.932 – Addressing data elements
(ASN.1 module according to ITU-T X.680/X.880) *(concluded)*

```
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),
                                    pTNSpecificNumber (3),
                                    localNumber (4),
                                    abbreviatedNumber (6)}

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

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

11) Subclause D.6.2 Indications related to flow control

A new Figure D.4 bis provides the coding of the Flow control component according to ITU-T X.680/X.880 and should be added after Figure D.4 as follows:

```
Flow-Control        {ccitt recommendation q 932 flow-control (8)}

        DEFINITIONS     ::=
        BEGIN
        EXPORTS         FlowControl;
        IMPORTS         OPERATION FROM Remote-Operations-Information-Objects
                        {joint-iso-ccitt (2) remote-operations (4) informationObjects(5) version1(0)};

        flowControl     OPERATION ::=
        {
        ARGUMENT SEQUENCE{
                ENUMERATED{
                        ReceiverNotReady (0),
                        ReceiverReady (1)},
                        MaximumNumberOfMessages [0] IMPLICIT INTEGER(1..255) OPTIONAL,
                        ReplenishmentParameter [1] IMPLICIT INTEGER(1..255) OPTIONAL,
                        TimerT [2] IMPLICIT INTEGER (1..30) OPTIONAL}
        CODE global: {ccitt recommendation q 932 flow-control (8) operation (1)}
                -- When the receiverReady coding is sent in FACILITY message, the
                -- maximumNumberOfMessages and timerT data elements may optionally
                -- be included. The MaximumNumberOfMessages is an integer that specifies
                -- the number of messages that may be sent in the time interval specifed by
                -- timerT for the call reference identified in the FACILITY message and these
                -- may optionally be included when the receiverReady coding is sent. The
                -- replenishmentParameter indicates the number of additional messages that
                -- may be sent at the end of timer interval timerT.
        }

-- This operation is a Class 5 operation
        END  -- of Flow Control
```

Figure D.4 *bis*/Q.932 – Flow Control Invoke Component ASN.1 structure

12) Appendix IV

A new Table IV.1 bis provides the definition of the ROSE protocol data types according to ITU-T X.680/X.880 and should be added after Table IV.12, together with the following note:

NOTE – This ASN.1 ROSE module is identical to the first module in Appendix A/X.880. The use of the bind and unbind operations is outside the scope of this Recommendation. They are only included to maintain the integrity of the referenced ROSE ASN.1 module.

Table IV.1 *bis*/Q.932 – Formal definitions of data types

```
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-itu-t 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,
                          &idempotent                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]
                          [IDEMPOTENT &idempotent]
                          [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]
                 }
                 }
```

Table IV.1 *bis*/Q.932 – Formal definitions of data types *(continued)*

```
OPERATION-PACKAGE ::=   CLASS
                        {
                                &Both                   OPERATIONOPTIONAL,
                                &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
                        }
```

Table IV.1 *bis*/Q.932 – Formal definitions of data types *(concluded)*

```
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
```

13) Subclause V.2 References

Add the following references:

[15] ITU-T Q.860 (2000), *ISDN and B-ISDN Generic Addressing and Transport (GAT) Protocol.*

[16] ITU-T X.680 (1997) | ISO/IEC 8824-1:1998, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*

[17] ITU-T X.880 (1994) | ISO/IEC 13712-1:1995, *Information technology – Remote Operations: Concepts, model and notation.*

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	**Switching and signalling**
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems

Geneva, 2001