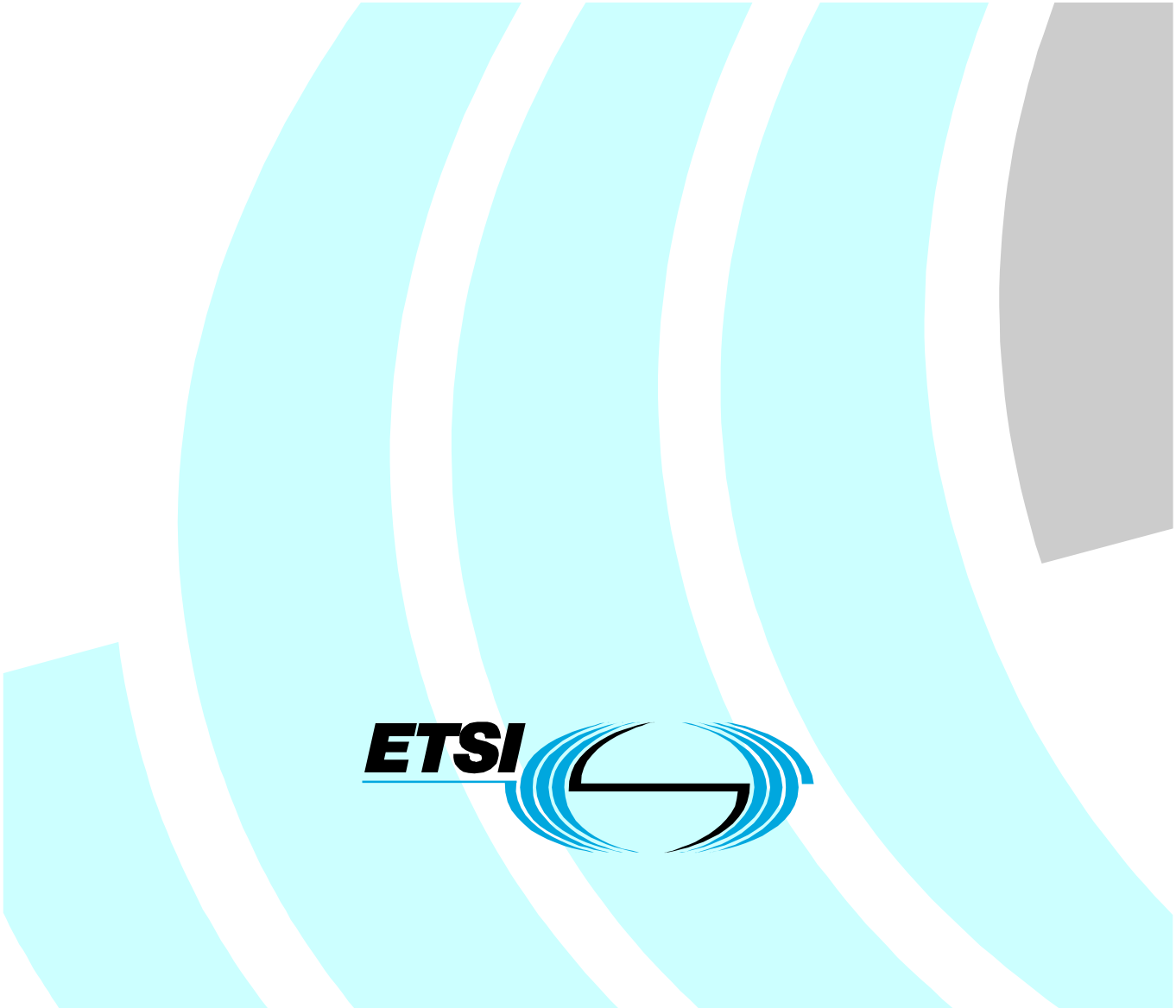
ETSI TS 102 052 V1.1.1 (2002-03)

Technical Specification

Telecommunications Management Network (TMN); TMN Broadband Access Coordination



Reference
DTS/TMN-00052

Keywords
management, TMN, VB5 interface

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Telecommunications Management Network (TMN).

1 Scope

The present document specifies the X interface between the Operations System (OS) of a Service Node (SN) and the Operations System (OS) of an Access Network (AN) for the coordination of the management associated with VB5.1 and VB5.2 traffic interfaces [1], [2] and the VB5 Q3 interfaces [3], [4].

Existing protocols are used where possible, and the focus of the work is on defining the object model. The definition of the functionality of TMN Operations Systems is outside the scope of the present document.

Security management is also outside the scope of the present document.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] ETSI EN 301 005-1: "V interfaces at the digital Service Node (SN); Interfaces at the VB5.1 reference point for the support of broadband or combined narrowband and broadband Access Networks (ANs); Part 1: Interface specification".

[2] ETSI EN 301 217-1: "V interfaces at the digital Service Node (SN); Interfaces at the VB5.2 reference point for the support of broadband or combined narrowband and broadband Access Networks (ANs); Part 1: Interface specification".

[3] ETSI EN 301 271: "Telecommunications Management Network (TMN); Management interfaces associated with the VB5.1 reference point".

[4] ETSI EN 301 754: "Telecommunications Management Network (TMN); Management interfaces associated with the VB5.2 reference point".

[5] ITU-T Recommendation G.902: "Framework Recommendation on functional access networks (AN) - Architecture and functions, access types, management and service node aspects".

[6] ITU-T Recommendation Q.832.3: "Broadband access coordination".

[7] ITU-T Recommendation X.721: "Information technology - Open Systems Interconnection - Structure of management information: Definition of management information".

[8] ITU-T Recommendation I.751: "Asynchronous transfer mode management of the network element view".

[9] ITU-T Recommendation M.3100: "Generic network information model".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in EN 301 005-1 [1], EN 301 217-1 [2], ITU-T Recommendation G.902 [5] and the following apply:

VB5 Resources: Management of user port functions and service port functions providing User Network Interface (UNI) and Service Node Interface (SNI) functionality, respectively, are considered in TS 102 052 based on the framework defined in ITU-T Recommendation G.902. Transmission specific resources lie outside its scope. VB5 Resources are referred to in TS 102 052 as resources.

3.2 Abbreviations

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

AN	Access Network
ASN.1	Abstract Syntax Notation one
ATM	Asynchronous Transfer Mode
B-BCC	Broadband Bearer Connection Control
GDMO	Guidelines for the Definition of Managed Objects
LUP	Logical User Port
OS	Operations System
RTMC	Real Time Management Coordination
SN	Service Node
SNI	Service Node Interface
TMN	Telecommunications Management Network
UNI	User Network Interface
VC	Virtual Channel
VP	Virtual Path
VPC	Virtual Path Connection
VPCI	Virtual Path Connection Identifier

4 General overview

The following information model diagrams have been drawn for the purpose of clarifying the relations between the different object classes of the model.

1) Entity-Relationship Models showing the relations of the different managed objects.

2) Inheritance Hierarchy showing how managed objects are derived from each other (i.e. the different paths of inherited characteristics of the different managed objects).

These diagrams are only for clarification. The formal specification in terms of GDMO templates and ASN.1 type definitions are the relevant information for implementations.

4.1 Entity-relationship models

The following conventions are used in the diagrams:

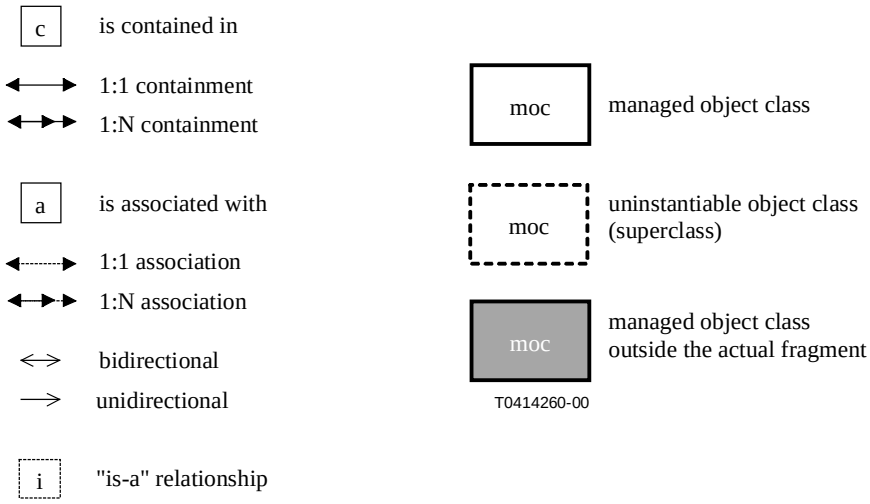


Figure 1: Conventions used in diagrams for Entity-Relationship Models

Where the directionality of containment is not clear it can be identified by implications since the root class is unique.

4.1.1 Entity-relationship diagram

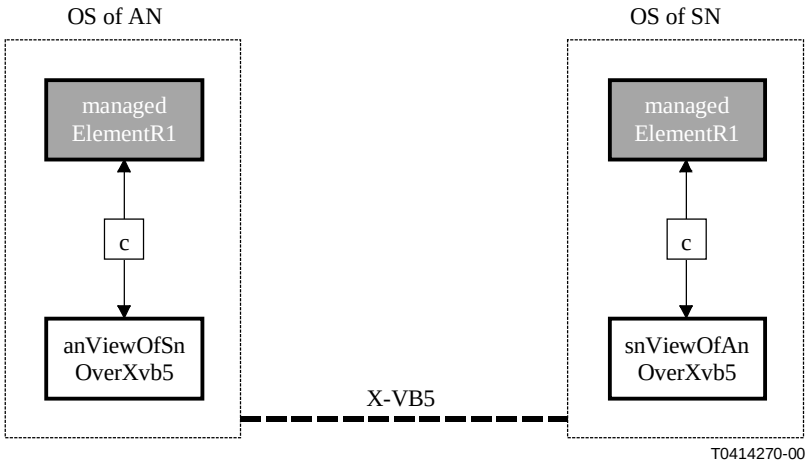


Figure 2: Entity-relationship diagram

4.2 Inheritance hierarchy

Figure 3 traces the inheritance relationships from the highest level object (see ITU-T Recommendation X.721 [7], "top") to the managed objects which are defined in the present document.

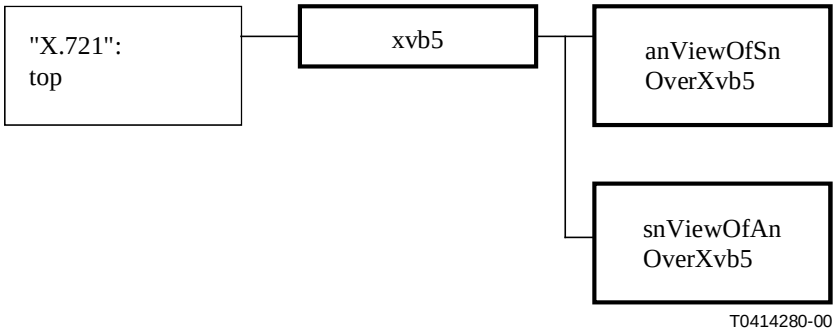


Figure 3: Inheritance Hierarchy

5 Formal object class definitions

This clause gives the formal definitions of the managed object classes, name bindings, behaviours, attributes, actions and notifications.

Formal definitions are shown in annex B.

6 Type definitions

Type definitions are shown in clause B.6.

7 Protocol stacks

The protocol stacks specified in ITU-T Recommendations Q.811, Q.812 and G.773 can be used as part of the protocol stack for the present document.

8 Conventions

Objects and their characteristics and associated ASN.1 defined here are given names with capitals used to indicate the start of the next word and acronyms are treated as if they were words.

Throughout the present document, all new attributes are named according to the following guidelines:

- The name of an attribute ends in the string "Ptr" if and only if the attribute value is intended to identify a single object.
- The name of an attribute ends in the string "PtrList" if and only if the attribute value is intended to identify one or more objects.
- The name of an attribute is composed of the name of an object class followed by the string "Ptr" if and only if the attribute value is intended to identify a specific object class.
- If an attribute is intended to identify different object classes, a descriptive name is given to that attribute and a description is provided in the attribute behaviour.

- The name of an attribute ends in the string "Id" if and only if the attribute value is intended to identify the name of an object, in which case this attribute should be the first one listed, should use ASN.1 NameType and should not be used to convey other information.

- The name of an attribute is composed of the name of an object class followed by the string "Id" if and only if the attribute value is intended to identify the name of the object class holding that attribute.

Annex A (normative): Management requirements

The management requirements are given below and in the provisioning principles for VB5.1 and VB5.2 interfaces.

A.1 General requirements

The general requirements include the general management coordination functions between the access network and the service node across Q3/X interfaces.

A.1.1 Coordinated VP and VC configuration

The configuration management function must support the coordinated addition and removal of VPs and VCs at both the UNIs and at the VB5 interfaces so that VP and VCs can be added and removed without disruption.

A.1.2 VPC checking

A mechanism is required to check the identity of VPCs which are set up between a user port and a service node so that mistakes in the cross-connection within an access network can be identified.

A.1.3 Coordination of port configuration data

The coordination of configuration information relating to user ports and service ports and their VPs and VCs is required to ensure consistency between the access network and the service node.

A.1.4 Coordination of VPCI values

There is a requirement for management coordination between the SN and the AN concerning the allocation of VPCI values for connections.

A.1.5 Consistency of configuration

There is a requirement to check the consistency of the configuration of logical user ports, logical service ports, and UNI accesses.

A.1.6 Availability of information

Information concerning a VB5 interface should not be visible to operators other than those related by that VB5 interface.

A.2 Coordination of the VB5 interface

A.2.1 Creation

There is a requirement to coordinate the creation of the VB5 interface, with the following information:

1) VP identifiers;
2) VCCs allocated for B-BCC and RTMC-protocols;
3) VCI range;
4) maximum number of simultaneously active VCCs;
5) maximum bandwidth of the VPC, provided by the traffic descriptor, on both directions (egress and ingress) which specifies e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
6) quality of service class of the VP;
7) other VCs provisioned in the VP;
8) the traffic profile of the VC, describing e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
9) quality of service class of the VC.

A.2.2 Verification and auditing

There is a requirement to verify and audit the correct configuration of the VB5 interface (for information involved see creation).

A.2.3 Modification

There is a requirement to coordinate the modification of the VB5 interface.

A.2.4 Deletion

There is a requirement to coordinate the deletion of the VB5 interface.

A.2.5 VC and VP provisioning

There is a requirement to coordinate the provisioning of VCs and VPs for users, with the following information:

1) VP/VC identifiers;
2) VCI range for VPs;
3) maximum number of simultaneously active VCCs in VP;
4) maximum bandwidth of the VPC, provided by the traffic descriptor, on both directions (egress and ingress) which specifies e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
5) quality of service class of the VP;
6) VCs provisioned in the VP;
7) the traffic profile of the VC, describing e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size.

A.2.6 Verification and auditing

There is a requirement to verify and audit the correct configuration of VCs and VPs for users (for information involved see provisioning).

A.2.7 VP and VC modification

There is a requirement to coordinate the modification of VCs and VPs for users.

A.2.8 Verification and auditing

There is a requirement to verify and audit the correct configuration of the VB5 interface (for information involved see creation).

A.2.9 VP and VC deletion

There is a requirement to coordinate the deletion of VCs and VPs for users.

A.3 Coordination of the UNI interface

A.3.1 Creation

There is a requirement to coordinate the creation of user VPs and user VCs between the users and the AN, with the following information:

1) VP/VC identifiers;
2) VCI range;
3) maximum number of simultaneously active VCCs;
4) maximum bandwidth of the VPC, provided by the traffic descriptor, on both directions (egress and ingress) which specifies e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
5) quality of service class of the VP;
6) the traffic profile of the VC, describing e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
7) quality of service class of the VC.

A.3.2 Verification and auditing

There is a requirement to verify and audit the correct configuration of user VPs and user VCs between the users and the AN (for information involved, see creation).

A.3.3 Modification

There is a requirement to coordinate the modification of user VPs and user VCs between the users and the AN.

A.3.4 Deletion

There is a requirement to coordinate the deletion of user VPs and user VCs between the users and the AN.

A.4 Broadband Bearer Connection Control coordination requirements

The requirements here are based on the need to support the VB5.2 B-BCC protocol [2].

The VPC/VCC at the VB5 and at the UNI interface may be selected both by the SN and by the AN, therefore both of them, the AN and the SN, must know the relevant information to do the CAC functions and select the right VPC/VC identifiers.

A.4.1 VCs at the VB5 interface

In order to support creation, deletion and modification of a VC at the VB5.2 interface under the control of B-BCC procedures the following information shall be available in the AN and the SN:

- the VPCs associated to the VB5.2 interface; the information associated to each VPC shall include:
  1) VCI range;
  2) maximum number of simultaneously active VCCs;
  3) maximum bandwidth of the VPC, provided by the traffic descriptor, on both directions (egress and ingress) which specifies e.g.:
     - peak cell rate;
     - cell delay variation tolerance;
     - sustainable cell rate;
     - maximum burst size;
  4) quality of service class of the VP;
  5) VCs allocated in the VP;
  6) the traffic profile of the VC, describing e.g.:
     - peak cell rate;
     - cell delay variation tolerance;
     - sustainable cell rate;
     - maximum burst size;
  7) quality of service class of the VC.

This information is available in the AN and SN, provided by the managed object classes modelling VP/VC connection points, defined in ITU-T Recommendation I.751 [88].

A.4.2 VCs at the UNI interface

In order to support creation, deletion and modification of VCs at the UNI interface under the control of B-BCC procedures the following information shall be available in the AN and the SN:

- the VPCs associated to the LUP; the information associated to each VPC shall include:
  1) VCI range;
  2) maximum number of simultaneously active VCCs;

3) maximum bandwidth of the VPC, provided by the traffic descriptor, on both directions (egress and ingress) which specifies e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
4) quality of service class of the VP;
5) VCs allocated in the VP;
6) the traffic profile of the VC, describing e.g.:
   - peak cell rate;
   - cell delay variation tolerance;
   - sustainable cell rate;
   - maximum burst size;
7) quality of service class of the VC.

This information is available in the AN, provided by the managed object classes modelling VP/VC connection points, defined in ITU-T Recommendation I.751 [88]. The same information shall be provided to the SN.

A.5 Fault and performance management

The following requirements have been identified for fault and performance management.

A.5.1 Fault reporting

Faults must be reported over the X interface when VB5 traffic interfaces are not operational.

A.5.2 Fault localization

The X interface must support the coordination activities that allow a Service Node and an Access Network to cooperate to locate the source of faults, for example when loopbacks are used.

Annex B (informative): Referenced definitions

This annex contains the referenced GDMO and ASN.1 definitions from ITU-T Recommendation Q.832.3 [6]. This is provided for convenience only and ITU-T Recommendation Q.832.3 [6] should be consulted for the normative text.

B.1 Object classes

B.1.1 anViewOfSnOverXvb5 (AN view of SN over X-VB5)

```
anViewOfSnOverXvb5 MANAGED OBJECT CLASS
    DERIVED FROM xvb5;
    CHARACTERIZED BY
        anViewOfSnOverXvb5Pkg PACKAGE
            BEHAVIOUR anViewOfSnOverXvb5Beh;
            ACTIONS
                anServiceLabelInquiry;;;
REGISTERED AS {q832-3ManagedObjectClass 1};

anViewOfSnOverXvb5Beh BEHAVIOUR
    DEFINED AS
        "This managed object represents the Service Node side of an X-VB5 interface, as seen by the
Operations System of the Access Network.";
```

B.1.2 snViewOfAnOverXvb5 (SN view of AN over X-VB5)

```
snViewOfAnOverXvb5 MANAGED OBJECT CLASS
    DERIVED FROM xvb5;
    CHARACTERIZED BY
        snViewOfAnOverXvb5Pkg PACKAGE
            BEHAVIOUR snViewOfAnOverXvb5Beh;
            ACTIONS
                addAnLoopRequest,
                snAccessLabelsInquiry,
                removeAnLoopRequest;;;
REGISTERED AS {q832-3ManagedObjectClass 2};

snViewOfAnOverXvb5Beh BEHAVIOUR
    DEFINED AS
        "This managed object represents the Access Network side of an X-VB5 interface, as seen by
the Operations System of the Service Node.";
```

B.1.3 xvb5 (X-VB5)

```
xvb5 MANAGED OBJECT CLASS
    DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
    CHARACTERIZED BY
        "ITU-T Recommendation M.3100":operationalStatePackage,
        xvb5Pkg PACKAGE
            BEHAVIOUR xvb5Beh;
            ATTRIBUTES
                xvb5Id
                    GET SET-BY-CREATE,
                " Rec. X.721 | ISO/IEC 10165-2": administrativeState
                    GET-REPLACE;
            ACTIONS
                addLupsRequest,
                addVb5ConnectionRequest,
                addVb5InterfaceRequest,
                addVb5ProtocolRequest,
                addVb5ProtocolVpRequest,
                addVb5VcsRequest,
                addVb5VpsRequest,
```

```
                auditVb5ConnectionRequest,
                auditVb5VpciRequest,
                listLupsRequest,
                listVb5ProtocolDetailsRequest,
                listVb5InterfacesRequest,
                listVb5VcsRequest,
                listVb5VpsRequest,
                removeLupsRequest,
                removeVb5ConnectionRequest,
                removeVb5InterfaceRequest,
                removeVb5ProtocolRequest,
                removeVb5ProtocolVpRequest,
                removeVb5VcsRequest,
                removeVb5VpsRequest;
            NOTIFICATIONS
                addLupsIndication,
                addVb5ConnectionIndication,
                addVb5InterfaceIndication,
                addVb5ProtocolIndication,
                addVb5ProtocolVpIndication,
                addVb5VcsIndication,
                addVb5VpsIndication,
                removeLupsIndication,
                removeVb5ConnectionIndication,
                removeVb5InterfaceIndication,
                removeVb5ProtocolIndication,
                removeVb5ProtocolVpIndication,
                removeVb5VcsIndication,
                removeVb5VpsIndication,
                resourceStatusIndication,
                " Rec. X.721 | ISO/IEC 10165-2": stateChange,
                " Rec. X.721 | ISO/IEC 10165-2": objectCreation,
                " Rec. X.721 | ISO/IEC 10165-2": objectDeletion;;;
REGISTERED AS {q832-3ManagedObjectClass 3};

xvb5Beh BEHAVIOUR
    DEFINED AS
        "The xvb5 managed object class represents the aspects of an X-VB5 interface that are common
to both sides. The xvb5 class is not instantiated.";
```

B.2 Name bindings

B.2.1 xvb5-managedElementR1

```
xvb5-managedElementR1 NAME BINDING
    SUBORDINATE OBJECT CLASS xvb5 AND SUBCLASSES;
    NAMED BY SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElementR1 AND SUBCLASSES;
    WITH ATTRIBUTE xvb5Id;
    CREATE
        WITH-REFERENCE-OBJECT,
        WITH-AUTOMATIC-INSTANCE-NAMING;
    DELETE
        DELETES-CONTAINED-OBJECTS;
REGISTERED AS {q832-3NameBinding 1};
```

B.3 Attributes

B.3.1 xvb5Id (X-VB5 identifier)

```
xvb5Id ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Q832-3ASN1Module.NameType;
    MATCHES FOR EQUALITY;
    BEHAVIOUR xvb5IdBeh;
REGISTERED AS {q832-3Attribute 1};
```

```
xvb5IdBeh BEHAVIOUR
    DEFINED AS
        "This attribute is used for naming instances of the managed object class xvb5 and
subclasses.";
```

B.4 Actions

B.4.1 addAnLoopRequest (add AN loop request)

```
addAnLoopRequest ACTION
    BEHAVIOUR addAnLoopRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddAnLoopRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddAnLoopRequestResult;
REGISTERED AS {q832-3Action 1};

addAnLoopRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used by the OS of the SN to request the OS of the AN to loop a connection so
that cells sent to the AN will be returned.";
```

B.4.2 addLupsRequest (add LUPs request)

```
addLupsRequest ACTION
    BEHAVIOUR addLupsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsRequestResult;
REGISTERED AS {q832-3Action 2};

addLupsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add Logical User Ports to a VB5 interface.";
```

B.4.3 addVb5ConnectionRequest (add VB5 connection request)

```
addVb5ConnectionRequest ACTION
    BEHAVIOUR addVb5ConnectionRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionRequestResult;
REGISTERED AS {q832-3Action 3};

addVb5ConnectionRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add a connection associated with a VB5
interface. The egress direction is out of the Access Network towards Service Node. The ingress
direction is into the Access Network from the Service Node.";
```

B.4.4 addVb5InterfaceRequest (add VB5 interface request)

```
addVb5InterfaceRequest ACTION
    BEHAVIOUR addVb5InterfaceRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceRequestResult;
REGISTERED AS {q832-3Action 4};

addVb5InterfaceRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add a VB5 interface.";
```

B.4.5 addVb5ProtocolRequest (add VB5 protocol request)

```
addVb5ProtocolRequest ACTION
    BEHAVIOUR addVb5ProtocolRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolRequestResult;
REGISTERED AS {q832-3Action 5};

addVb5ProtocolRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add a protocol to an existing VB5
interface.";
```

B.4.6 addVb5ProtocolVpRequest (add VB5 protocol VP request)

```
addVb5ProtocolVpRequest ACTION
    BEHAVIOUR addVb5ProtocolVpRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpRequestResult;
REGISTERED AS {q832-3Action 6};

addVb5ProtocolVpRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add a protocol VP to a VB5 interface.";
```

B.4.7 addVb5VcsRequest (add VB5 VCs request)

```
addVb5VcsRequest ACTION
    BEHAVIOUR addVb5VcsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsRequestResult;
REGISTERED AS {q832-3Action 7};

addVb5VcsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add VCs to a VP which is associated with a
VB5 interface.";
```

B.4.8 addVb5VpsRequest (add VB5 VPs request)

```
addVb5VpsRequest ACTION
    BEHAVIOUR addVb5VpsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsRequestResult;
REGISTERED AS {q832-3Action 8};

addVb5VpsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to add VPs that are associated with a VB5
interface.";
```

B.4.9 anServiceLabelInquiry (AN service label inquiry)

```
anServiceLabelInquiry ACTION
    BEHAVIOUR anServiceLabelInquiryBeh;
    MODE CONFIRMED;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AnServiceLabelInquiryResult;
REGISTERED AS {q832-3Action 9};

anServiceLabelInquiryBeh BEHAVIOUR
    DEFINED AS
        "This action is used by the OS of an AN to inquire the label that an SN uses for the AN.";
```

B.4.10 auditVb5ConnectionRequest (audit VB5 connection request)

```
auditVb5ConnectionRequest ACTION
    BEHAVIOUR auditVb5ConnectionRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5ConnectionRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5ConnectionRequestResult;
REGISTERED AS {q832-3Action 10};

auditVb5ConnectionRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to audit a connection which is associated with a
VB5 interface.";
```

B.4.11 auditVb5VpciRequest (audit VB5 VPCI request)

```
auditVb5VpciRequest ACTION
    BEHAVIOUR auditVb5VpciRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5VpciRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5VpciRequestResult;
REGISTERED AS {q832-3Action 11};

auditVb5VpciRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to audit a VPCI which is associated with a VB5
interface.";
```

B.4.12 listLupsRequest (list LUPs request)

```
listLupsRequest ACTION
    BEHAVIOUR listLupsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListLupsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListLupsRequestResult;
REGISTERED AS {q832-3Action 12};

listLupsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to list the Logical User Ports associated with a
VB5 interface between an Access Network and a Service Node which the two Operations Systems together
control.";
```

B.4.13 listVb5ProtocolDetailsRequest (list protocol details request)

```
listVb5ProtocolDetailsRequest ACTION
    BEHAVIOUR listVb5ProtocolDetailsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5ProtocolDetailsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5ProtocolDetailsRequestResult;
REGISTERED AS {q832-3Action 13};

listVb5ProtocolDetailsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to list the details of the protocols of a VB5
interface between an Access Network and a Service Node which the two Operations Systems together
control.";
```

B.4.14 listVb5InterfacesRequest (list VB5 interfaces request)

```
listVb5InterfacesRequest ACTION
    BEHAVIOUR listVb5InterfacesRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5InterfacesRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5InterfacesRequestResult;
```

```
REGISTERED AS {q832-3Action 14};

listVb5InterfacesRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to list the identities of the VB5 interfaces
between Access Network(s) and the Service Node(s) which the two Operations Systems together
control.";
```

B.4.15 listVb5VcsRequest (list VB5 VCs request)

```
listVb5VcsRequest ACTION
    BEHAVIOUR listVb5VcsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VcsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VcsRequestResult;
REGISTERED AS {q832-3Action 15};

listVb5VcsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to list the VCs associated with a VB5
interface.";
```

B.4.16 listVb5VpsRequest (list VB5 VPs request)

```
listVb5VpsRequest ACTION
    BEHAVIOUR listVb5VpsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VpsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VpsRequestResult;
REGISTERED AS {q832-3Action 16};

listVb5VpsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to list the VPs associated with a VB5
interface.";
```

B.4.17 removeAnLoopRequest (remove AN loop request)

```
removeAnLoopRequest ACTION
    BEHAVIOUR removeAnLoopRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveAnLoopRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveAnLoopRequestResult;
REGISTERED AS {q832-3Action 17};

removeAnLoopRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used by the OS of the SN to request the OS of the AN to remove a loop from a
connection.";
```

B.4.18 removeLupsRequest (remove LUPs request)

```
removeLupsRequest ACTION
    BEHAVIOUR removeLupsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsRequestResult;
REGISTERED AS {q832-3Action 18};

removeLupsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove Logical User Ports from a VB5
interface.";
```

B.4.19 removeVb5ConnectionRequest (remove VB5 connection request)

```
removeVb5ConnectionRequest ACTION
    BEHAVIOUR removeVb5ConnectionRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionRequestResult;
REGISTERED AS {q832-3Action 19};

removeVb5ConnectionRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove a connection associated with a VB5
interface. The egress direction is out of the Access Network towards Service Node. The ingress
direction is into the Access Network from the Service Node.";
```

B.4.20 removeVb5InterfaceRequest (remove VB5 interface request)

```
removeVb5InterfaceRequest ACTION
    BEHAVIOUR removeVb5InterfaceRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceRequestResult;
REGISTERED AS {q832-3Action 20};

removeVb5InterfaceRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove a VB5 interface.";
```

B.4.21 removeVb5ProtocolRequest (remove VB5 protocol request)

```
removeVb5ProtocolRequest ACTION
    BEHAVIOUR removeVb5ProtocolRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolRequestResult;
REGISTERED AS {q832-3Action 21};

removeVb5ProtocolRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove a protocol from a VB5 interface.";
```

B.4.22 removeVb5ProtocolVpRequest (remove VB5 protocol Vp request)

```
removeVb5ProtocolVpRequest ACTION
    BEHAVIOUR removeVb5ProtocolVpRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpRequestResult;
REGISTERED AS {q832-3Action 22};

removeVb5ProtocolVpRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove the protocol VP from a VB5
interface.";
```

B.4.23 removeVb5VcsRequest (remove VB5 VCs request)

```
removeVb5VcsRequest ACTION
    BEHAVIOUR removeVb5VcsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsRequestResult;
```

```
REGISTERED AS {q832-3Action 23};

removeVb5VcsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove VCs from a VP which is associated with
a VB5 interface.";
```

B.4.24 removeVb5VpsRequest (remove VB5 VPs request)

```
removeVb5VpsRequest ACTION
    BEHAVIOUR removeVb5VpsRequestBeh;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsRequestInfo;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsRequestResult;
REGISTERED AS {q832-3Action 24};

removeVb5VpsRequestBeh BEHAVIOUR
    DEFINED AS
        "This action is used to request the peer OS to remove VPs that are associated with a VB5
interface.";
```

B.4.25 snAccessLabelsInquiry (SN access labels inquiry)

```
snAccessLabelsInquiry ACTION
    BEHAVIOUR snAccessLabelsInquiryBeh;
    MODE CONFIRMED;
    WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.SnAccessLabelsInquiryResult;
REGISTERED AS {q832-3Action 25};

snAccessLabelsInquiryBeh BEHAVIOUR
    DEFINED AS
        "This action is used by the OS of an SN to inquire the access labels that an AN uses for the
SN and the VB5 interface.";
```

B.5 Notifications

B.5.1 addLupsIndication (add LUPs indication)

```
addLupsIndication NOTIFICATION
    BEHAVIOUR addLupsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsIndicationInfo;
REGISTERED AS {q832-3Notification 1};

addLupsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of Logical User Ports to a
VB5 interface.";
```

B.5.2 addVb5ConnectionIndication (add VB5Connection indication)

```
addVb5ConnectionIndication NOTIFICATION
    BEHAVIOUR addVb5ConnectionIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionIndicationInfo;
REGISTERED AS {q832-3Notification 2};

addVb5ConnectionIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of a connection associated
with a VB5 interface. The egress direction is out of the Access Network towards Service Node. The
ingress direction is into the Access Network from the Service Node.";
```

B.5.3 addVb5InterfaceIndication (add VB5 interface indication)

```
addVb5InterfaceIndication NOTIFICATION
    BEHAVIOUR addVb5InterfaceIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceIndicationInfo;
REGISTERED AS {q832-3Notification 3};

addVb5InterfaceIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to inform the peer OS that a new VB5 interface has been added.";
```

B.5.4 addVb5ProtocolIndication (add VB5 protocol indication)

```
addVb5ProtocolIndication NOTIFICATION
    BEHAVIOUR addVb5ProtocolIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolIndicationInfo;
REGISTERED AS {q832-3Notification 4};

addVb5ProtocolIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of a protocol to an
existing VB5 interface.";
```

B.5.5 addVb5ProtocolVpIndication (add VB5 protocol VP indication)

```
addVb5ProtocolVpIndication NOTIFICATION
    BEHAVIOUR addVb5ProtocolVpIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpIndicationInfo;
REGISTERED AS {q832-3Notification 5};

addVb5ProtocolVpIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of a protocol VP to a VB5
interface.";
```

B.5.6 addVb5VcsIndication (add VB5 VCs indication)

```
addVb5VcsIndication NOTIFICATION
    BEHAVIOUR addVb5VcsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsIndicationInfo;
REGISTERED AS {q832-3Notification 6};

addVb5VcsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of VCs to a VP which is
associated with a VB5 interface.";
```

B.5.7 addVb5VpsIndication (add VB5 VPs indication)

```
addVb5VpsIndication NOTIFICATION
    BEHAVIOUR addVb5VpsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsIndicationInfo;
REGISTERED AS {q832-3Notification 7};

addVb5VpsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the addition of VPs that are associated
with a VB5 interface.";
```

B.5.8 removeLupsIndication (remove LUPs indication)

```
removeLupsIndication NOTIFICATION
    BEHAVIOUR removeLupsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsIndicationInfo;
REGISTERED AS {q832-3Notification 8};

removeLupsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of Logical User Ports from a
VB5 interface.";
```

B.5.9 removeVb5ConnectionIndication (remove VB5Connection indication)

```
removeVb5ConnectionIndication NOTIFICATION
    BEHAVIOUR removeVb5ConnectionIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionIndicationInfo;
REGISTERED AS {q832-3Notification 9};

removeVb5ConnectionIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of a connection associated
with a VB5 interface. The egress direction is out of the Access Network towards Service Node. The
ingress direction is into the Access Network from the Service Node.";
```

B.5.10 removeVb5InterfaceIndication (remove VB5 interface indication)

```
removeVb5InterfaceIndication NOTIFICATION
    BEHAVIOUR removeVb5InterfaceIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceIndicationInfo;
REGISTERED AS {q832-3Notification 10};

removeVb5InterfaceIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of a VB5 interface.";
```

B.5.11 removeVb5ProtocolIndication (remove VB5 protocol indication)

```
removeVb5ProtocolIndication NOTIFICATION
    BEHAVIOUR removeVb5ProtocolIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolIndicationInfo;
REGISTERED AS {q832-3Notification 11};

removeVb5ProtocolIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of a protocol from a VB5
interface.";
```

B.5.12 removeVb5ProtocolVpIndication (remove VB5 protocol VP indication)

```
removeVb5ProtocolVpIndication NOTIFICATION
    BEHAVIOUR removeVb5ProtocolVpIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpIndicationInfo;
REGISTERED AS {q832-3Notification 12};

removeVb5ProtocolVpIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of the protocol VP from a
VB5 interface.";
```

B.5.13 removeVb5VcsIndication (remove VB5 VCs indication)

```
removeVb5VcsIndication NOTIFICATION
    BEHAVIOUR removeVb5VcsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsIndicationInfo;
REGISTERED AS {q832-3Notification 13};

removeVb5VcsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of VCs from a VP which is
associated with a VB5 interface.";
```

B.5.14 removeVb5VpsIndication (remove VB5 VPs indication)

```
removeVb5VpsIndication NOTIFICATION
    BEHAVIOUR removeVb5VpsIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsIndicationInfo;
REGISTERED AS {q832-3Notification 14};

removeVb5VpsIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the removal of VPs that are associated
with a VB5 interface.";
```

B.5.15 resourceStatusIndication (resource status indication)

```
resourceStatusIndication NOTIFICATION
    BEHAVIOUR resourceStatusIndicationBeh;
    WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ResourceStatusIndicationInfo;
REGISTERED AS {q832-3Notification 15};

resourceStatusIndicationBeh BEHAVIOUR
    DEFINED AS
        "This notification is used to notify the peer OS of the change of status of a Resource.";
```

B.6 Type definitions

```
Q832-3ASN1DefinedTypesModule {
        itu-t (0) recommendation (0) q (17) q832 (832) dot (127) coord (3)
            informationModel (0) asn1Modules (2) asn1DefinedTypesModule (0)}

DEFINITIONS IMPLICIT TAGS ::=

BEGIN
-- EXPORTS everything

IMPORTS
    ObjectInstance
    FROM CMIP-1 {joint-iso-itu-t ms (9) cmip (1) modules (0) protocol (3)}
NameType
    FROM ASN1DefinedTypesModule {ccitt recommendation m 3100
        informationModel(0) asn1Modules(2) asn1DefinedTypesModule(0)}
    VciValue, VpiValue
    FROM AtmMIBMod {itu-t (0) recommendation (0) i (9) atmm (751)
        informationModel (0) asn1Module (2) atm(0)}
    VpciValue
    FROM AtmMIBMod {itu-t (0) recommendation (0) q (17) 824 (824) dot (127) bsm (6)
        informationModel (0) asn1Modules (2) asn1DefinedTypesModule (0)}
;  -- end of imports

-- start of object identifier definitions

q832-3InformationModel
    OBJECT IDENTIFIER ::=
    {itu-t (0) recommendation (0) q (17) q832 (832) dot (127) coord(3) informationModel(0)}
q832-3StandardSpecificExtension
    OBJECT IDENTIFIER ::= {informationModel q832-3StandardSpecificExtension(0)}
q832-3ManagedObjectClass
    OBJECT IDENTIFIER ::= {informationModel q832-3ManagedObjectClass(3)}
```

```
q832-3Package
    OBJECT IDENTIFIER ::= {informationModel q832-3Package(4)}
q832-3NameBinding
    OBJECT IDENTIFIER ::= {informationModel q832-3NameBinding(6)}
q832-3Attribute
    OBJECT IDENTIFIER ::= {informationModel q832-3Attribute (7)}
q832-3Action
    OBJECT IDENTIFIER ::= {informationModel q832-3Action(9)}
q832-3Notification
    OBJECT IDENTIFIER ::= {informationModel q832-3Notification(10)}

-- end of object identifier definitions

-- other ASN1 definitions in alphabetical order - all these are new

AddAnLoopRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalUserPortNumber   [1] INTEGER OPTIONAL,
    vpciValue       [2] VpciValue,
    vciValue        [3] VciValue OPTIONAL}

AddAnLoopRequestResult ::= CHOICE {
    loopAdded   [0] NULL,
    loopNotAdded    [1] LoopNotAddedInfo}

AddLupsIndicationInfo ::= AddLupsRequestInfo

AddLupsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalUserPortNumber   [1] SEQUENCE OF INTEGER}

AddLupsRequestResult ::= INTEGER {
    lupAdded    (0),
    lupNotAdded     (1)}

AddVb5ConnectionIndicationInfo ::= AddVb5ConnectionRequestInfo

AddVb5ConnectionRequestInfo ::= SEQUENCE {
    egressPeakCellRateCLP0Plus1     [0] INTEGER,
    egressPeakCellRateCLP0          [1] INTEGER,
    ingressPeakCellRateCLP0Plus1        [2] INTEGER,
    ingressPeakCellRateCLP0         [3] INTEGER,
    egressSustainableCellRateCLP0Plus1  [4] INTEGER,
    egressSustainableCellRateCLP0       [5] INTEGER,
    ingressSustainableCellRateCLP0Plus1 [6] INTEGER,
    ingressSustainableCellRateCLP0      [7] INTEGER,
    egressCDVToleranceCLP0Plus1     [8] INTEGER,
    egressCDVToleranceCLP0      [9] INTEGER,
    ingressCDVToleranceCLP0Plus1        [10]    INTEGER,
    ingressCDVToleranceCLP0     [11]    INTEGER,
    egressMaxBurstSizeCLP0Plus1     [12]    INTEGER,
    egressMaxBurstSizeCLP0          [13]    INTEGER,
    ingressMaxBurstSizeCLP0Plus1        [14]    INTEGER,
    ingressMaxBurstSizeCLP0     [15]    INTEGER,
    egressQosClass              [16]    INTEGER,
    ingressQosClass             [17]    INTEGER,
    vciValueA               [18]    INTEGER OPTIONAL,
    vciValueB               [19]    INTEGER OPTIONAL,
    physicalPortA               [20]    INTEGER,
    vpiValueA               [21]    INTEGER,
    logicalServicePortA         [22]    INTEGER OPTIONAL,
    vpciValueA              [23]    INTEGER OPTIONAL,
    physicalPortB               [24]    INTEGER,
    vpiValueB               [25]    INTEGER,
    logicalServicePortB         [26]    INTEGER OPTIONAL,
    vpciValueB              [27]    INTEGER OPTIONAL}

AddVb5ConnectionRequestResult ::= INTEGER {
    vb5ConnectionAdded  (0),
    vb5ConnectionNotAdded   (1)}

AddVb5InterfaceIndicationInfo ::= AddVb5InterfaceRequestInfo

AddVb5InterfaceRequestInfo ::= INTEGER  -- Logical service port number
```

```
AddVb5InterfaceRequestResult ::= INTEGER {
    vb5InterfaceAdded   (0),
    vb5InterfaceNotAdded    (1)}

AddVb5ProtocolIndicationInfo ::= AddVb5ProtocolRequestInfo

AddVb5ProtocolRequestInfo ::= SEQUENCE {
    logicalServicePortNumber        [0] INTEGER,
    vpciValue           [1] VpciValue,
    maxNumVciBitsNearEnd        [2] INTEGER OPTIONAL,
    maxNumVciBitsSupported  [3] INTEGER OPTIONAL,
    maxNumActiveVccsAllowed [4] INTEGER OPTIONAL,
    maxNumActiveVccsNearEnd [5] INTEGER OPTIONAL}

AddVb5ProtocolRequestResult ::= INTEGER {
    vb5ProtocolAdded    (0),
    vb5ProtocolNotAdded (1)}

AddVb5ProtocolVpIndicationInfo ::= AddVb5ProtocolVpRequestInfo

AddVb5ProtocolVpRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    vpciValue       [1] VpciValue,
    vpProfile       [2] VpProfile OPTIONAL}

AddVb5ProtocolVpRequestResult ::= INTEGER {
    vb5ProtocolVpAdded  (0),
    vb5ProtocolVpNotAdded   (1)}

AddVb5VcsIndicationInfo ::= AddVb5VcsRequestInfo

AddVb5VcsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalServiceSubport   [1] INTEGER OPTIONAL,
    vpciValue       [2] VpciValue,
    vciValue        [3] VciValue}

AddVb5VcsRequestResult ::= INTEGER {
    vb5VcsAdded     (0),
    vb5VcsNotAdded  (1)}

AddVb5VpsIndicationInfo ::= AddVb5VpsRequestInfo

AddVb5VpsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalServiceSubport   [1] INTEGER OPTIONAL,
    addVpInfo       [2] AddVpInfo}

AddVb5VpsRequestResult ::= INTEGER {
    vb5VpsAdded     (0),
    vb5VpsNotAdded  (1)}

AddVpInfo ::= SEQUENCE OF SEQUENCE {
    physicalPort        [0] INTEGER,
    vpiValue        [1] VpiValue,
    vpciValue       [2] VpciValue,
vpProfile       [3] VpProfile OPTIONAL}

AnServiceLabelInquiryResult ::= INTEGER OPTIONAL

AuditVb5ConnectionRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    specifiedVpOrVc     [1] SpecifiedVpOrVc}

AuditVb5ConnectionRequestResult ::= CHOICE {
    connectionAudited   [0] SpecifiedVpOrVc,
    connectionNotAudited    [1] ConnectionNotAuditedInfo}

AuditVb5VpciRequestInfo ::= SEQUENCE {
    logicalServicePortNumber        [0] INTEGER,
    specifiedNniVpciOrRemoteVp  [1] SpecifiedNniVpciOrRemoteVp}

AuditVb5VpciRequestResult ::= CHOICE {
    vpciAudited [0] SpecifiedNniVpciOrRemoteVp,
    vpciNotAudited  [1] VpciNotAuditedInfo}
```

```
ConnectionNotAuditedInfo ::= INTEGER {
    unspecified     (0),
    unknownPhysicalPort (1),
    unknownVpiValue (2),
    unknownVciValue (3)}

ListLupsRequestInfo ::= INTEGER -- Logical service port number

ListLupsFailureInfo ::= INTEGER {
    unspecified     (0),
    unknownLspNumber    (1)}

ListLupsRequestResult ::= CHOICE {
    success [0] INTEGER,    -- Logical user port number
    failure [1] ListLupsFailureInfo}

ListVb5ProtocolDetailsRequestInfo ::= INTEGER   -- Logical service port number

ListVb5ProtocolDetailsRequestResult ::= CHOICE {
    success [0] ListVb5ProtocolDetailsSuccessInfo,
    failure [1] ListVb5ProtocolDetailsFailureInfo}

ListVb5ProtocolDetailsFailureInfo ::= INTEGER {
    unspecified         (0),
    unknownLspNumber        (1)}

ListVb5ProtocolDetailsSuccessInfo ::= SEQUENCE OF Vb5ProtocolDetails

Vb5ProtocolDetails ::= SEQUENCE {
    protocolType                [0] ProtocolType,
    vpciValue               [1] VpciValue,
    vciValue                [2] VciValue,
    egressPeakCellRateCLP0Plus1     [3] INTEGER,
    egressPeakCellRateCLP0          [4] INTEGER,
    ingressPeakCellRateCLP0Plus1        [5] INTEGER,
    ingressPeakCellRateCLP0         [6] INTEGER,
    egressSustainableCellRateCLP0Plus1  [7] INTEGER,
    egressSustainableCellRateCLP0       [8] INTEGER,
    ingressSustainableCellRateCLP0Plus1 [9] INTEGER,
    ingressSustainableCellRateCLP0      [10]    INTEGER,
    egressCDVToleranceCLP0Plus1     [11]    INTEGER,
    egressCDVToleranceCLP0      [12]    INTEGER,
    ingressCDVToleranceCLP0Plus1        [13]    INTEGER,
    ingressCDVToleranceCLP0     [14]    INTEGER,
    egressMaxBurstSizeCLP0Plus1     [15]    INTEGER,
    egressMaxBurstSizeCLP0          [16]    INTEGER,
    ingressMaxBurstSizeCLP0Plus1        [17]    INTEGER,
    ingressMaxBurstSizeCLP0     [18]    INTEGER,
    bufferRelease               [19]    BOOLEAN,
    maxCc                   [20]    INTEGER,
    maxInformationFieldLength       [21]    INTEGER,
    maxLengthSscopUuField           [22]    INTEGER,
    maxPd                   [23]    INTEGER,
    maxSscopCreditToPeer            [24]    INTEGER,
    maxStat                 [25]    INTEGER,
    sscopTimerCc                [26]    INTEGER,
    sscopTimerIdle              [27]    INTEGER,
    sscopTimerKeepAlive         [28]    INTEGER,
    sscopTimerNoResponse            [29]    INTEGER,
    sscopTimerPoll              [30]    INTEGER}

ProtocolType::= INTEGER {
    RTMC            (1),
    BBCC            (2)}

ListVb5InterfacesRequestInfo ::= SEQUENCE OF ObjectInstance

ListVb5InterfacesRequestResult ::= SEQUENCE OF INTEGER  -- Logical service port number

ListVcsFailureInfo ::= INTEGER {
    unspecified         (0),
    unknownLspNumber        (1),
    unknownLupNumber        (2),
    unknownVpciValue        (3),
    unknownLupVpciCombination   (4)}
```

```
ListVb5VcsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber        [0] INTEGER,
    logicalUserPortNumber       [1] INTEGER OPTIONAL,
    vpciValue           [2] VpciValue OPTIONAL}

ListVb5VcsRequestResult ::= CHOICE {
    success [0] ListVcsSuccessInfo,
    failure [1] ListVcsFailureInfo}

ListVcsSuccessInfo ::= SEQUENCE {
    logicalUserPortNumber   [0] INTEGER OPTIONAL,
    vpciValue       [1] VpciValue,
    vciValue        [2] VciValue}

ListVb5VpsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalUserPortNumber   [1] INTEGER OPTIONAL}

ListVb5VpsRequestResult ::= CHOICE {
    success [0] ListVb5VpsSuccessInfo,
    failure [1] ListVb5VpFailureInfo}

ListVb5VpFailureInfo ::= INTEGER {
    unspecified         (0),
    unknownLspNumber        (1),
    unknownLupNumber        (2)}

ListVb5VpsSuccessInfo ::= SEQUENCE OF Vb5VpDetails

Vb5VpDetails ::= SEQUENCE {
    logicalUserPortNumber       [0] INTEGER OPTIONAL,
    physicalPort            [1] INTEGER,
    vpiValue            [2] INTEGER,
    vpciValue           [3] INTEGER OPTIONAL,
    maxNumVciBitsNearEnd        [4] INTEGER OPTIONAL,
    maxNumVciBitsSupported  [5] INTEGER OPTIONAL,
    maxNumActiveVccsAllowed [6] INTEGER OPTIONAL,
    maxNumActiveVccsNearEnd [7] INTEGER OPTIONAL}

LoopNotAddedInfo ::= INTEGER {
    unspecified         (0),
    unknownLogicalServicePort   (1),
    unknownLogicalUserPort      (2),
    unknownVpciValue        (3),
    unknownVciValue     (4),
    loopAlreadyPresent      (5)}

LoopNotRemovedInfo ::= INTEGER {
    unspecified         (0),
    unknownLogicalServicePort   (1),
    unknownLogicalUserPort      (2),
    unknownVpciValue        (3),
    unknownVciValue     (4),
    noLoopPresent           (5)}

RemoveAnLoopRequestInfo ::= AddAnLoopRequestInfo

RemoveAnLoopRequestResult ::= CHOICE {
    loopRemoved     [0] NULL,
    loopNotRemoved       [1]    LoopNotRemovedInfo}

RemoveLupsIndicationInfo ::= RemoveLupsRequestInfo

RemoveLupsRequestInfo ::= AddLupsRequestInfo

RemoveLupsRequestResult ::= INTEGER {
    lupRemoved (0),
    lupNotRemoved (1)}

RemoveVb5ConnectionIndicationInfo ::= RemoveVb5ConnectionRequestInfo

RemoveVb5ConnectionRequestInfo ::= SEQUENCE {
    vciValueA               [0] INTEGER OPTIONAL,
    vciValueB               [1] INTEGER OPTIONAL,
    physicalPortA               [2] INTEGER,
    vpiValueA               [3] INTEGER,
    logicalServicePortA         [4] INTEGER OPTIONAL,
    vpciValueA              [5] INTEGER OPTIONAL,
```

```
    physicalPortB               [6] INTEGER,
    vpiValueB               [7] INTEGER,
    logicalServicePortB         [8] INTEGER OPTIONAL,
    vpciValueB              [9] INTEGER OPTIONAL}

RemoveVb5ConnectionRequestResult ::= INTEGER {
    vb5ConnectionRemoved (0),
    vb5ConnectionNotRemoved (1)}

RemoveVb5InterfaceIndicationInfo ::= RemoveVb5InterfaceRequestInfo

RemoveVb5InterfaceRequestInfo ::= AddVb5InterfaceRequestInfo

RemoveVb5InterfaceRequestResult ::= INTEGER {
    vb5InterfaceRemoved (0),
    vb5InterfaceNotRemoved (1)}

RemoveVb5ProtocolIndicationInfo ::= RemoveVb5ProtocolRequestInfo

RemoveVb5ProtocolRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    vb5ProtocolType     [1] Vb5ProtocolType}

RemoveVb5ProtocolRequestResult ::= INTEGER {
    vb5ProtocolRemoved (0),
    vb5ProtocolNotRemoved (1)}

RemoveVb5ProtocolVpIndicationInfo ::= RemoveVb5ProtocolVpRequestInfo

RemoveVb5ProtocolVpRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    vpciValue       [1] VpciValue}

RemoveVb5ProtocolVpRequestResult ::= INTEGER {
    vb5ProtocolVpRemoved (0),
    vb5ProtocolVpNotRemoved (1)}

RemoveVb5VcsIndicationInfo ::= RemoveVb5VcsRequestInfo

RemoveVb5VcsRequestInfo:: = AddVb5VcsRequestInfo

RemoveVb5VcsRequestResult ::= INTEGER {
    vb5VcsRemoved (0),
    vb5VcsNotRemoved (1)}

RemoveVb5VpsIndicationInfo ::= RemoveVb5VpsRequestInfo

RemoveVb5VpsRequestInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalServiceSubport   [1] INTEGER OPTIONAL,
    removeVpInfo        [2] RemoveVpInfo}

RemoveVb5VpsRequestResult ::= INTEGER {
    vb5VpsRemoved (0),
    vb5VpsNotRemoved (1)}

RemoveVpInfo ::= SEQUENCE OF SEQUENCE {
    physicalPort        [0] INTEGER,
    vpiValue        [1] VpiValue,
    vpciValue       [2] VpciValue}

ResourceStatusIndicationInfo ::= SEQUENCE {
    logicalServicePortNumber    [0] INTEGER,
    logicalUserPortNumber   [1] INTEGER OPTIONAL,
    vpciValue       [2] VpciValue,
    resourceStatus      [3] ResourceStatus}

ResourceStatus ::= INTEGER {
    fullyOperational                (0),
    administratelyBlockedTestCallsAllowed   (1),
    administratelyBlockedNoCellFlow     (2),
    fault                   (3)}

SnAccessLabelsInquiryResult ::= SEQUENCE {
    snLabel     [0] INTEGER OPTIONAL,
    interfaceLabel  [0] INTEGER OPTIONAL}
```

```
SpecifiedNniVpci ::= SEQUENCE {
    logicalUserPortNumber   [0] INTEGER OPTIONAL,
    vpciValue       [1] VpciValue}

SpecifiedNniVpciOrRemoteVp ::= CHOICE {
    specifiedNniVpci        [0] SpecifiedNniVpci,
    specifiedRemoteVp   [1] SpecifiedVp}

SpecifiedVc ::= SEQUENCE {
    physicalPort    [0] INTEGER,
    vpiValue    [1] VpiValue,
    vciValue    [2] VciValue}

SpecifiedVp ::= SEQUENCE {
    physicalPort    [0] INTEGER,
    vpiValue    [1] VpiValue}

SpecifiedVpOrVc ::= CHOICE {
    specifiedVp [0] SpecifiedVp,
    specifiedVc [1] SpecifiedVc}

Vb5ProtocolType ::= INTEGER {
    rtmc (0),
    bbcc (1)}

VpciNotAuditedInfo ::= INTEGER {
    unspecified (0),
    unknownLupNumber (1),
    unknownVpci (2),
    unknownPhysicalPort (3),
    unknownVpiValue (4)}

VpProfile ::= SEQUENCE {
    maxNumVciBitsNearEnd        0]  INTEGER,
    maxNumVciBitsSupported  [1] INTEGER,
    maxNumActiveVccsAllowed [2] INTEGER,
    maxNumActiveVccsNearEnd [3] INTEGER}

END – of Q832-3ASN1DefinedTypesModule
```

Annex C (informative): Bibliography

- ITU-T Recommendation G.773: "Protocol suites for Q-interfaces for management of transmission systems".
- ITU-T Recommendation G.774: "SDH Management Information Model".
- ITU-T Recommendation G.803: "Architecture of Transport Networks based on the Synchronous Digital Hierarchy (SDH)".
- ITU-T Recommendation I.211: "B-ISDN service aspects".
- ITU-T Recommendation I.311: "B-ISDN General Network Aspects".
- ITU-T Recommendation I.327: "B-ISDN functional architecture".
- ITU-T Recommendation I.356: "B-ISDN ATM Layer Cell Transfer Performance".
- ITU-T Recommendation I.371: "Traffic control and congestion control in B-ISDN".
- ITU-T Recommendation I.413: "B-ISDN user-network interface".
- ITU-T Recommendation I.432: "B-ISDN user-network interface - Physical layer specification".
- ITU-T Recommendation I.580: "General arrangements for interworking between B-ISDN and 64 kbit/s based ISDN".
- ITU-T Recommendation I.610: "B-ISDN Operation and Maintenance principles and functions".
- ITU-T Recommendation I.732: "Functional characteristics of ATM equipment".
- ITU-T Recommendation M.3200: "TMN Management Services".
- ITU-T Recommendation M.3207.1: "TMN management service: Maintenance aspects of B-ISDN management".
- ITU-T Recommendation M.3400: "TMN Management Functions".
- ITU-T Recommendation M.3610: "Principles for applying the TMN concept to the management of B-ISDN".
- ITU-T Recommendation Q.811: "Lower layer protocol profiles for the Q3 and X interfaces".
- ITU-T Recommendation Q.812: "Upper layer protocol profiles for the Q3 and X interfaces".
- ITU-T Recommendation Q.821: "Stage 2 and stage 3 description for the Q3 interface - Alarm Surveillance".
- ITU-T Recommendation Q.822: "Stage 1, Stage 2 and Stage 3 Description for the Q3 interface performance management".
- ITU-T Recommendation X.680: "Specification of Abstract Syntax Notification One (ASN.1)".
- ITU-T Recommendation X.701: "System Management Overview".
- ITU-T Recommendation X.722: "Guidelines for Definition of Managed Objects".
- ITU-T Recommendation X.733: "Information technology - Open systems interconnection - Systems management: Alarm Reporting Function".
- ITU-T Recommendation X.734: "Information technology - Open systems interconnection - Systems management: Event report management function".
- ITU-T Recommendation X.735: "Information technology - Open systems interconnection - Systems management: Log Control Functions".

- ITU-T Recommendation X.737 : "Information technology - Open systems interconnection - Systems management: Confidence and Diagnostic Test Function".
- ITU-T Recommendation X.738: "Information technology - Open systems interconnection - Systems management: Summarization function".
- ITU-T Recommendation X.739: "Information technology - Open systems interconnection - Systems management: Metric Objects and Attributes".
- ITU-T Recommendation X.745: "Information technology - Open systems interconnection - Systems management: Test management function".
- ITU-T Recommendation X.746: "Information technology - Open systems interconnection - Systems management: Scheduling function".
- ATM Forum Specification: "M4 Interface Requirements and Logical MIB: ATM Network Element View, Version 1.0".
- ATM Forum Specification: "CMIP Specification for the M4 Interface, Version 1.0".
- ATM Forum Specification: "ATM User-Network Interface Specification, Version 3.0".
- ATM Forum Specification: "ATM User-Network Interface Specification, Version 3.1".

History

Document history		
V1.1.1	March 2002	Publication