Final draft EN 300 820-1 V1.1.1 (1998-04)

European Standard (Telecommunications series)

Network Aspects (NA); Management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) cross connected network; Part 1: Configuration management aspects

European Telecommunications Standards Institute

Reference

DEN/NA-052212-1 (7lc90ico.PDF)

Keywords

ATM, B-ISDN, management, TMN

ETSI Secretariat

Postal address

F-06921 Sophia Antipolis Cedex - FRANCE

Office address

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

Internet

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

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 ETR 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available **free of charge** from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.fr/ipr or http://www.etsi.org/ipr).

Pursuant to the ETSI Interim 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 ETR 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This European Standard (Telecommunication series) has been produced by ETSI Technical Committee Network Aspects (NA), and is now submitted for the Voting phase of the ETSI standards Two-step Approval Procedure.

The present document is part 1 of a multi-part EN covering the management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) cross connected network, as identified below:

Part 1: "Configuration management aspects";

Part 2: "Asynchronous Transfer Mode (ATM) VP alarm management";

Part 3: "Performance management aspects".

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

1 Scope

The present document addresses the requirements of network, and service providers of Asynchronous Transfer Mode (ATM) cross connected networks for establishing, maintaining and releasing Virtual Path (VP) Virtual Channel (VC) connections, which span several administrative ATM domains. These requirements are satisfied by the use of a standardized interface (the "X-interface") between Operation Systems (OSs) belonging to different network operators.

The present document contains a general overview describing the different management areas that will be covered in the different X-interface ENs - configuration, fault and performance - as well as the relationships between them.

The present document describes the configuration management area covering the following aspects:

- a management architecture that shows how the X-interface is to be used between service or network providers;
- the management services and functions needed to manage ATM connections, which span several administrative domains. These management services and functions cover the requirements for the X-interface;
- the management information crossing the X-interface. This management information specification uses the Guidelines for the Definition of Managed Objects (GDMO) formalism, described in ITU-T Recommendation X.722 [6].

2 Normative references

References may be made to:

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

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

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

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

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

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

[2] ITU-T Recommendation X.721 | ISO 10165-2: "Information technology - Open Systems Interconnection - Structure of management information: Definition of management information".

[3] I-ETS 300 653 (1996): "Telecommunications Management Network (TMN); Generic managed object class library for the network level view".

[4] ITU-T Recommendation M.1400: "Designations for international networks".

[5] ITU-T Recommendation I.751 (03/96): "Asynchronous transfer mode management of the network element view".

[6] ITU-T Recommendation X.722: "Information technology - Open Systems Interconnection - Structure of Management information: Guidelines for the definition of managed objects".

[7] EN 300 820-2: "Telecommunications Management Networks (TMN); Management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP) / Virtual Channel (VC) cross connected network; Part 2: Fault management aspects".

3 Definitions and abbreviations

3.1 Definitions

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

(Some definitions depend on the future acceptance of the "cascaded/mixed mode" as described in annex D. This dependence is already taken into account in these definitions.)

A Public Network Operator (PNO): The A PNO is the PNO whose subnet is connected to the A User. The A PNO can be the Initiating one, but this is not always the case. It is the Consumer of other PNO's parts of the VP / VC connection.

connection: A "transport entity" which is capable of transferring information transparently between "Connection Points (CP)". A "connection" defines the association between the "CPs" and the "CPs" delimit the "connection".

consumer and provider roles of a PNO: With respect to a particular VP / VCC, a Consumer PNO is a PNO that has delegated the management of a VP / VC subnetwork connection plus the outgoing link connection (both shall be part of the connection) to another PNO (being a Provider PNO). If, in future, the „cascaded/mixed" mode should be accepted (annex D) a PNO can have both roles at once, if it is providing part of the VP / VCC (being a Provider), and at the same time asks another PNO to provide a part of the connection (being a Consumer).

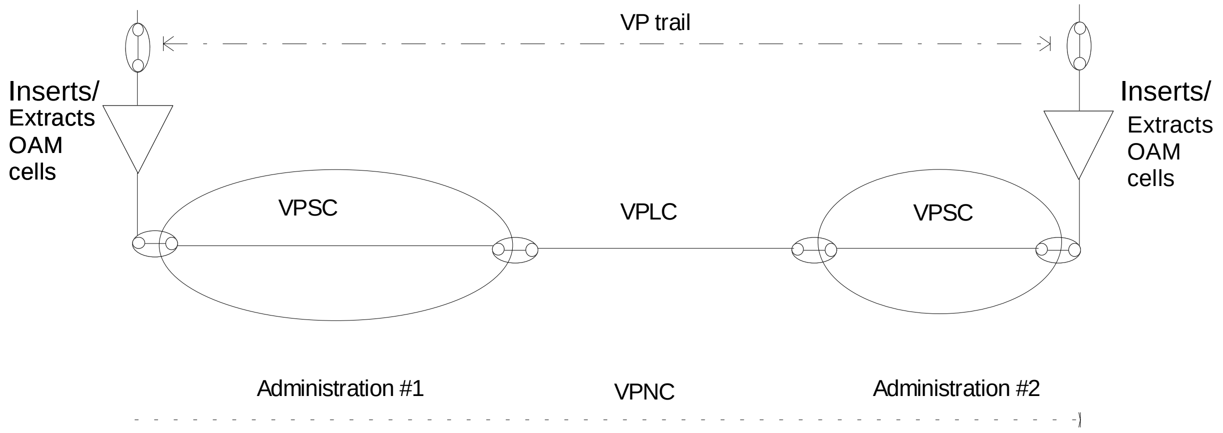
initiating Public Network Operator (PNO): The Initiating PNO is the PNO requesting for a particular ATM connection starting in the subnetwork of the A side; It controls the overall VP / VC connection.

link connection: A "transport entity" provided by the "client/server" association. It is formed by a near-end "adaptation" function, a server "trail" and a far-end "adaptation" function between "CPs". It can be configured as part of the "trail management process" in the associated server layer.

link: A "topological component" which describes the fixed relationship between a "sub-network" and another "sub-network" or "access group".

network connection: A "transport entity" formed by the series of "connections" between "termination CPs".

sub-network connection: A "transport entity" formed by a "connection" across a "sub-network" between "CPs". It can be configured as part of the "trail management process".



VPLC: VP Link Connection
VPNC: VP Network Connection
VPSC: VPSub Network Connection

Figure 1: Functional architecture of a VPC provided by two administrations

subnetwork view: A subtree of the X-interface tree. A subnetwork view belongs to a particular PNO: The PNO at the root of the subtree.

sub-network: A "topological component" used to effect routing and management. It describes the potential for "sub-network connections" across the "sub-network". It can be partitioned into interconnected "sub-networks" and "links". Each " sub-network" in turn can be partitioned into smaller "sub-networks " and "links" and so on. A "sub-network" may be contained within one physical node.

transit PNO: A Transit PNO is a PNO using its own subnetwork to perform its required transit part of VP / VC connection. It has a provider role and corresponds to a leaf in the X-interface tree, not being the **Z** side. In the "cascaded/mixed approach" case it can be both a Provider (where it acts as a Transit) and a Consumer (where it virtually acts as an Initiating).

X-interface tree: With respect to a particular VP / VCC, X-interface relations exist between each Provider PNO and their Consumer PNO. Because each Provider has exactly one Consumer, the X-interface relations between all PNOs involved in the management of a particular VP / VCC form a tree, the X-interface relation tree.

NOTE: For a particular VP / VCC there can be several possible X-interface relation trees; the actual tree is formed at VP / VCC setup. The root of the tree is the Initiating PNO; it uses (and via an X-interface controls) the PNOs (often Transit PNOs), to which it is connected in the tree via its branches. The most right leaf of the tree is the Z PNO. Figure 2 shows an example of a X-interface tree.

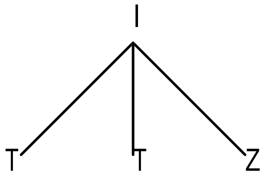


Figure 2: Example of a X-interface tree with the Initiating PNO being the A PNO

Z PNO: A Z PNO is a PNO whose subnet is connected to the Z User. It has a Provider role and corresponds to the rightmost leaf in the X-interface tree and connects the Z User to the reservation. In the „cascaded/mixed" approach case it can be both a Provider (where it acts as the Z side) and a Consumer (where it virtually acts as an Initiating PNO).

3.2 Abbreviations

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

ASN.1	Abstract Syntax Notation One
ATM	Asynchronous Transfer Mode
BM	Business Management
CMIP	Common Management Information Protocol
CMISE	Common Management Information Service Element
CP	Connection Point
CTP	Connection Termination Point
EM	Element Management
GDMO	Guidelines for the Definition of Managed Objects
GOM	Generic Object Model
MF	Management Function
MS	Management Service
MSC	Management Service Component
NE	Network Element
NEF	Network Element Function
NM	Network Management
NMS	Network Management System
OS	Operations System
OSF	Operations System Function
PNO	Public Network Operator
QoS	Quality of Service
SM	Service Management
TMN	Telecommunications Management Network
TP	Termination Point
VC	Virtual Channel

VCC	Virtual Channel Connection
VP	Virtual Path
VPC	Virtual Path Connection
VPLC	Virtual Path Link Connection
VPNC	Virtual Path Network Connection
VPSC	Virtual Path Subnetwork Connection

4 Management architecture

This clause describes the functional architecture for the X-interface for ATM cross connected networks. It provides an informative overview of the use of the X-interface and is therefore not normative. (Some definitions partially depend on the future acceptance of the „cascaded/mixed mode" as described in annex D. This dependence is already taken into account.)

Three concepts underpin the functional architecture for the X-interface:

- The X-interface connects two management systems, for the purpose of exchanging service level and/or network level requests with each other.
- Consumer/Provider roles (also reflected by Manager/Agent).
- The future use of Star or Cascaded organizational models for communication, or a mixture of both. The choice of the organizational model will be determined by agreements between the PNOs involved in the X-interface.

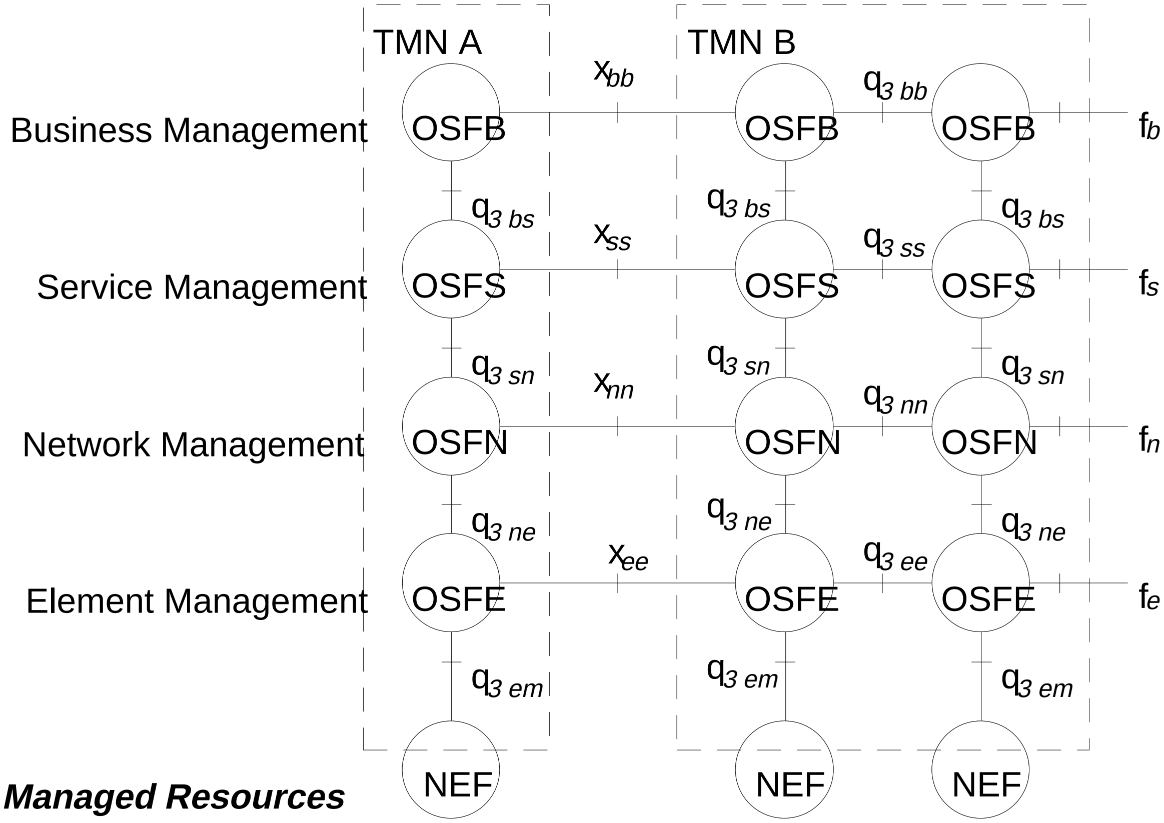
These concepts are elaborated in the following subclauses.

4.1 X-interface at the Service Management level

In order to clarify the position of the X-interface within the layered management architecture outlined in ITU-T Recommendation M.3100 [1], the following definitions are adopted within the present document (figure 3):

- The Network Management (NM) level is concerned with connections within the network. This means the control of topological information (subnetworks and the links between subnetworks), and subnetwork connections.
- The Service Management (SM) level is concerned with the overall connection and its associated Quality of Service (QoS).

Since Network operators can request other network operators to deliver a connection with a certain QoS, over the X-interface, this interface can be considered at the SM level. However, some functionalities described in the present document are allocated to the NM level, such as the management of topological information.



q3-em: between NE and EM level

q3-ne: between EM and NM level

q3-sn: between SM and NM level

q3-bs: between BM and SM level

q3/x-nn in between two NM levels

q3/x-ss in between two SM levels

q3/x-bb in between two BM levels

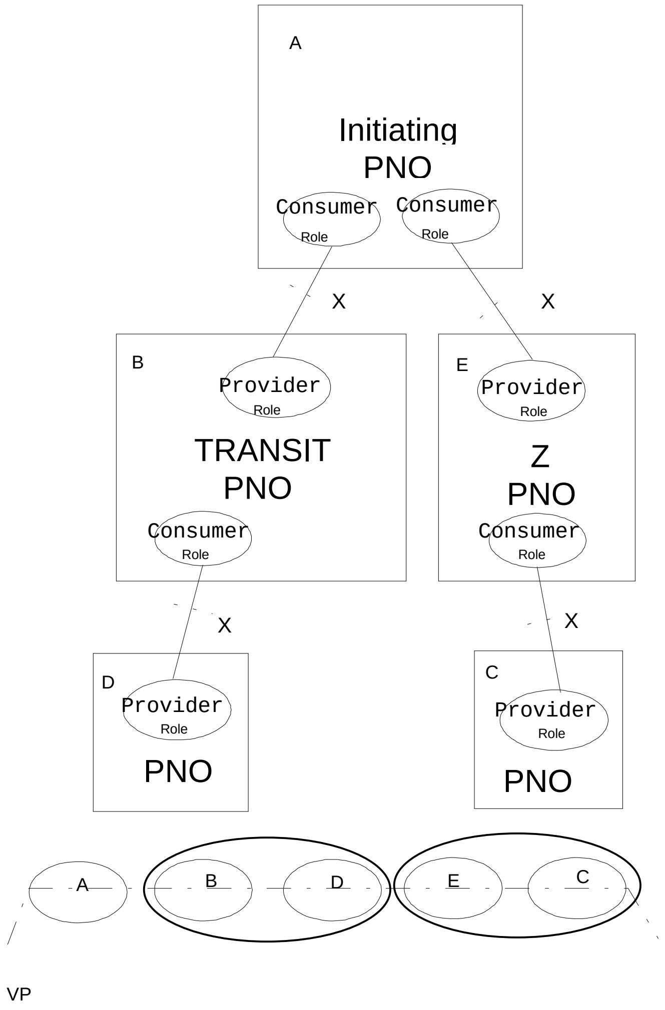
q3/x-ee in between two EM levels

f-e, f-n, f-s, f-b: between OS functionality and workstations

Figure 3: Layers of management (from ITU-T Recommendation M.3100 [1])

4.2 Consumer/Provider

The functional architecture for the X-interface for ATM Cross Connected networks is characterized by Consumer/Provider roles: each PNO in a VP / VC Connection is a Provider of a part of that connection. However, if the "cascaded/mixed mode" will be accepted, some PNOs might also have a Consumer role, since they use parts of other PNO's networks to provide their part of the connection. This is illustrated in figure 4.



PNO = Public Network Operator
VP = Virtual Path

Figure 4: Consumer/provider roles over an X-interface for a specific connection

4.3 Organizational model

This subclause describes the organizational model, which will be used in establishing X-interface relationships between PNOs. It is shown in figure 5.

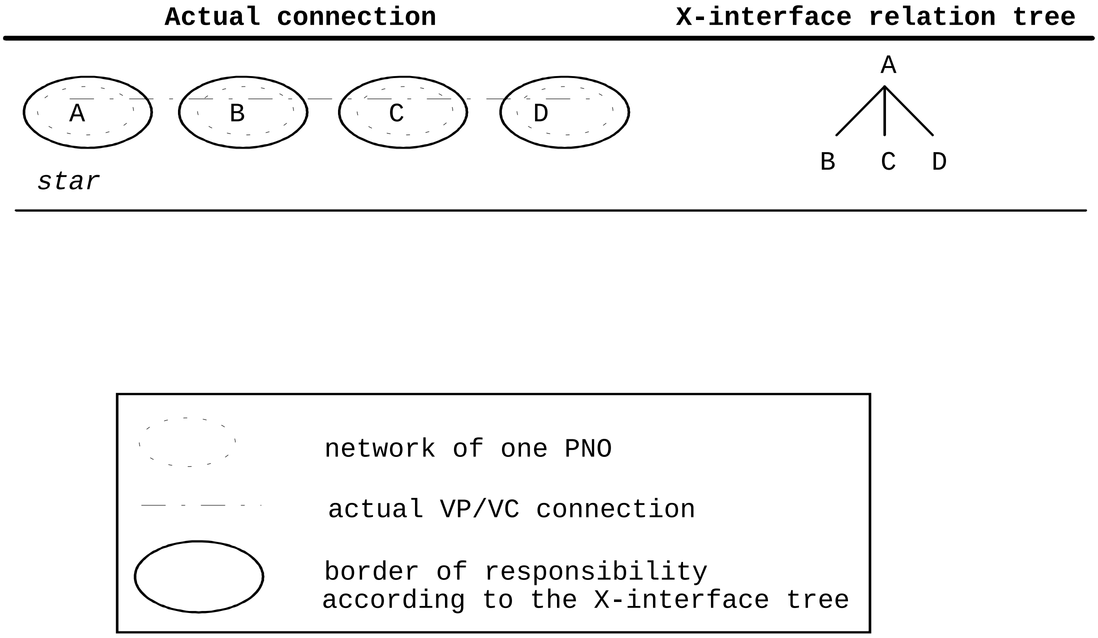


Figure 5: Organizational model

In the *Star* organization as exemplified in the figure above, PNO A uses the X-interface relation with all the PNOs involved. This means that PNO A has full responsibility for the entire connection and all the other PNOs are only responsible for their own network towards PNO A.

The Consumer/Provider roles, introduced in subclause 4.2, are reflected by the X-interface tree: in the *Star* example above, PNO A has a Consumer role, whereas PNO B, C and D have a Provider role.

4.4 Responsibility of PNOs regarding a VP / VC connection

This subclause denotes which responsibility is required from each PNO involved a particular VP / VC connection. The following rules apply:

Rule 1 A PNO is responsible for the management of a particular VP / VCC within its own PNO Subnetwork (if not being the ***Initiating PNO,*** who is responsible towards the connection customer).

Rule 2 Besides the responsibility in rule 1, a PNO is also responsible for the ATM Connection over the physical connection from its PNO Subnetwork to the next PNO Subnetwork on the route of a VP / VCC (seen from the ***A*** network towards the ***Z*** network).

4.4.1 Examples of application of the responsibility rules

The responsibility rules described in subclause 4.4 are illustrated by the following scenario description. The scenario starts with the existence of a particular VP / VCC between PNOs A and C via B and a particular X-interface relation tree. The scenario describes the occurrence of 4 failures that need management action; each of them is described in one step of the scenario description. The 2 steps are an example of part of the life cycle of this particular VP / VCC and take place in sequence. The scenario description is illustrated in figures 6 through 7.

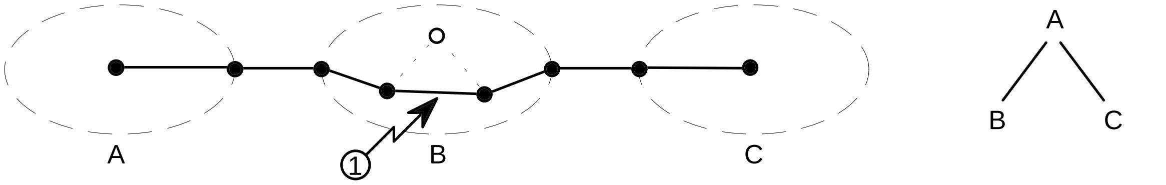


Figure 6: Step 1 of example scenario

Step 1

A fault has occurred in the VP / VCSC within the PNO subnetwork of PNO B. PNO B reports its Consumer PNO (PNO A) about the failure and indicates that the failure is recoverable. PNO B starts the reconfiguration process and notifies its superior about this (for suppressing alarms). After reconfiguration has been successfully completed, PNO B reports this to its Consumer.

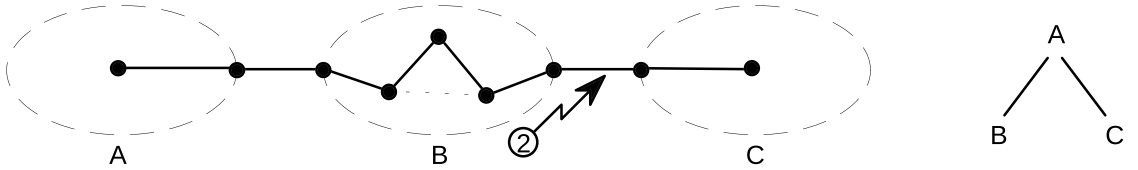


Figure 7: Step 2 of example scenario

Step 2

A fault has occurred in the inter-PNO VP / VCLC between PNO B and PNO C. Since this is the outgoing inter-PNO VP / VCLC of the PNO subnetwork of PNO B, PNO B is responsible for reporting this failure to its Consumer PNO (PNO A). PNO B reports the failure to PNO A and indicates that it can recover the failure. PNO B performs protection switching on the faulty inter-PNO VP / VCLC and notifies its Consumer about this (for suppressing alarms) and also informs him about the Id of the new physical link in the connection.

5 General overview of the X-interface specification

The specification of the X-type Interface between Operation Systems of a VP / VC cross connected network is split into three ENs, each one dealing with a specific management area:

- EN 300 820-1: "Network Aspects (NA); Management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) cross connected network;
  Part 1: Configuration management aspects".
- EN 300 820-2 [7]: "Telecommunications Management Networks (TMN); Management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) cross connected network; Part 2: Fault management aspects".
- EN 300 820-3: "Telecommunications Management Networks (TMN); Management information model for the X-type interface between Operation Systems (OSs) of a Virtual Path (VP)/Virtual Channel (VC) cross connected network; Part 3: Performance management aspects".

6 Requirements

Two categories of requirements are presented: general and configuration management.

In cases where requirements are reused from other sources, our own interpretation of those requirements is included.

6.1 General requirements

The following general requirements apply for the management functionality for the X-interface for ATM cross-connected networks:

G1. All transactions can be subject to authentication and Access Control. Other security services like Data Integrity, Confidentiality and Non-repudiation might be desirable.

G2. In order to achieve unambiguous identification of entities that are manageable, assigning unique names is necessary.

G3. The X-interface is solely concerned with the management of the network and the VP / VC Bearer Service.

G4. The X-interface for a given PNO may assume either the provider or the consumer role, dependant on the connection.

G5. Message transfer should be accomplished in real time.

G6. Notification of management of X-interface failure is for further study.

G7. Some definition of behaviour is required in order to avoid deadlock situations across the X-interface. For further study.

G8. It should be possible to manage a VP / VC Connection (including the establishment and release of a VPC/ VCC), through a third party PNO. However a PNO has the right to refuse a connection, in which the Initiating PNO is not taking part.

NOTE: Since the X-interface specification caters for interworking between Telecommunications Management Network (TMN) domains, it may be necessary to define domain types, e.g. Public TMN and Private TMN.

6.2 Configuration management requirements

C1. It should be possible to indicate in the path setup requests:

- Quality of Service;
- Preferred routes (where a route is a sequence of PNO networks);
- Physical separation of routes (end-to-end based , connection based or sub network based);
- Traffic description;
- Group Identification;
- Connection Reference.

NOTE: Other items for path setup request are for further study.

C2. It should be possible to verify whether the delivered path conforms to the setup request. This also tracing of a route of a connection. Also, following the reconfiguration of a connection it may be required to determine the actual route (down to PNO subnetwork & inter-PNO connections) of an active connection.

C3. Data representing the physical and logical resources comprising the subnetwork CPs will need to be maintained. An example of a physical resource is the physical link and the data representing it is physical link identifier. An example of a logical resource is a VP and the data representing it is the VPI.

C4. Both on-request and scheduled path configuration (reservation, cancellation, activation, deactivation) should be supported by the X-interface. Scheduled connection state change will be required in cases when multiple connections have to be brought into service in a predetermined sequence.

C5. Schedule modification (including activation/deactivation schedules). Once established, a schedule for . connection state changes may need to be modified.

C6. Cell rate modification on an active connection. It should be possible to modify the data rate on an established active connection without the need to deactivate the connection.

C7. It shall be possible to re-negotiate the QoS at any phase of connection lifecycle. QoS parameters may be altered by user request or PNO intervention.

C8. Grouping of subnetwork connections in accordance with user defined criteria. Connections may need to be grouped according to criteria other than their ***A*/*Z***. For example, users may want to refer to a mixed group of VP and VC connections as a single unit.

C9. It shall be possible to request that a new connection take the same route as an existing connection group or connection.

C10. Cancellation, activation and deactivation operations may be performed on connection groups.

7 Management services

7.1 Management services

For the Configuration Management area, the following Management Service (MS) is identified:

VP / VC Service Provisioning - Consists on the provisioning of User-to-User VP / VC Connections.

The VP / VC Service Provisioning MS is decomposed into four Management Service Components (MSCs):

- *Establish MSC* - It allows the reservation and activation of a VPC/ VCC.
- *Release MSC* - It allows the cancellation of a reservation or the deactivation of a VPC/ VCC.
- *Modification MSC* - It allows the modification of the characteristics of a VPC/ VCC, during reservation time, e.g. the cell rate.
- *Reconfiguration MSC* - It allows the reconfiguration of the whole or a part of a VPC/ VCC, when a unrecoverable failure is detected.

7.2 Management Service Components (MSC)

7.2.1 Establish MSC

Summary Description

Having received a request for a VP / VC Connection from the service layer, a check (*Destination User Checking-request*) is carried out with the ***Z*** PNO to ensure that the ***Z*** side wishes to accept the requested connection. If the response to this check (*Destination User Checking-response*) is positive, then another check is made with the ***Z*** side to see if there is sufficient cell rate available on at least one incoming link to the ***Z*** PNO (*Check Available Cell Rate-request*). If the response to the check (*Check Available Cell Rate-response*) is positive, then possible paths will be found between the ***A*** and ***Z***, using the topology information . Both of these procedures are optional, in the sense that the Initiating PNO is not obliged to perform them in order to establish a VP / VC Connection.

(A *Check Available Cell Rate-request* can also be made with a transit PNO)

The topology information that describe the inter-PNO links have been made available by the other PNOs either because they sent it (Topology Info Changes-request) or because it was read from the PNOs MIBs.

Once a path has been selected, each PNO involved in the connection is serially requested to perform a reservation (*Reserve VP / VC Subnetwork Connection-request*) between a specified input link and a specified adjacent subnetwork. The PNO responds (*Reserve VP / VC Subnetwork Connection-response*) with the result of this reservation request.

If all PNOs involved in the connection return positive results then the connection can be activated in accordance with the scheduled time specified in the reservation.

If by any reason, a given PNO Subnetwork do not activate the VP / VC Connection in accordance with the schedule, the Initiating PNO will issue an activation request (*Activate VP / VC Subnetwork Connection-request*) to this PNO. This one shall then respond (*Activate VP / VC Subnetwork Connection-response*) with the result of the activation. However, this does not guarantee that the connection is operational.

The activation procedures will also be used in the case of performing tests, e.g. continuity check, before the real scheduled activation. The initiation of these procedures is up to the Initiating PNO.

If a reservation or activation request fails then each PNO who has already reserved or activated the connection needs to be told to cancel their resource allocation for this VP / VC Connection (*Cancel VP / VC Subnetwork Connection-request*).

As a background task the topology of the network is being tracked and any changes to either PNO Subnetworks or inter-PNO links are sent out to all PNOs (*Topology Info Changes-request*).

List of Functions

- Destination User Checking *(Optional)*;
- Check Available Cell Rate *(Optional)*;
- Reserve VP / VC Subnetwork Connection;
- Activate VP / VC Subnetwork Connection;
- Cancel VP / VC Subnetwork Connection;
- Topology Info Changes (This is a background function that will provide network topology and status information to be used during the establishment process).

7.2.2 Release MSC

Summary Description

A VP / VC connection can be released for several reasons including expiry of the duration, a service layer request or due to reconfiguration.

There are two parts to the release of a VP / VC connection, deactivation of the VP / VC connection, e.g. for a scheduled release, and release of the resources (*Release VP / VC Subnetwork Connection).*

The deactivation is made in accordance with the schedule of the reservation. A explicit deactivation of the VP / VC Connection by the Initiating PNO has to be made in the following circumstances:

- failure of a PNO Subnetwork to deactivate a VP / VC Subnetwork Connection;
- rerouting of a VP / VC Connection;
- in case of activation for test purposes outside the scheduled time slots.

The release of the reservation is made by the Initiating PNO by means of issuing a request - *Release VP / VC Subnetwork Connection-request* - to the other PNOs involved in the connection to release their portion of the appropriate connection and the PNOs will each respond (*Release VP / VC Subnetwork Connection-response*) indicating the result of the requested release. This release of the resources may also be required if a reservation or activation fails at any PNO.

A release request can also be received while a connection is active; it will be released correctly.

When a zPNO has to release a subnetwork connection in his subnetwork the Initiating PNO will be notified (cancel VP / VC Subnetwork Connection Notification). The Initiating PNO then can release the connection or reconfigure it.

If the initiating PNO does not release the connection the zPNO can release it himself and send a *delete Notification* to the initiating PNO.

List of Functions

- Deactivate VP / VC Subnetwork Connection;
- Cancel VP / VC Subnetwork Connection (Notification);
- Release VP / VC Subnetwork Connection.

7.2.3 Modification MSC

Summary Description

Modification is allowed during the reservation time (interval between the VP / VC Connection creation and deletion) of a VP / VC connection. Changes are possible for the scheduling and the cell rate a connection. The path may not be changed while altering the connection.

To perform the Modification of a VPC/ VCC , the Initiating PNO requests in parallel, each PNO involved in the connection, to perform a change reservation *(Change Reservation-request)*, by reserving the new values of the parameters, without cancelling the old ones for its own subnetwork connection. Each PNO responds with the result of this change reservation request *(Change Reservation-response)*.

If all PNOs involved in the connection return positive results, then the modified connection can be activated by sending an *Activate Change-request* in parallel to each PNO, which will then respond with the result of the activation *(Activate Change-response)*.

If a change reservation request fails, then each PNO which has already reserved the changes needs to be told to cancel the altered resource allocation for this VP / VC connection, and to keep the old one *(Cancel Change-request)*.

If an activate change request fails, then a cancel reservation to each involved PNO is issued by the Initiating PNO *(Cancel VP / VC Subnetwork Connection-request).*

List of Functions

- Change Reservation;
- Cancel Change;
- Activate Change;
- Cancel VP / VC Subnetwork Connection.

7.2.4 Reconfiguration MSC

Summary Description

Reconfiguration via the X-interface shall be done by the Initiating PNO (I-PNO), after an unrecoverable failure at the VP / VC Subnetwork Connections , which are part of the VP / VCC.

Reconfiguration in this case means Rerouting, which is the establishment of a replacement connection by the Network Management System (NMS) function.

For reconfiguring a VP / VC connection, it will be necessary to find a route around the faulty subnetwork of the current VP / VC connection.

Reconfiguration at the Intiating PNO level consists of Cancelling the Reservation and make a new Establishment using the old parameters. The reconfiguration process can occur at any time.

If the inter_PNO management of the VP / VC connection is organized using the *star* approach, reconfiguration of a VP / VC connection boils down to deactivating and cancelling the appropriate VP / VC subnetwork connections (*Deactivate VP / VC Subnetwork Connection* and *Release VP / VC Subnetwork Connection*), followed by reserving and activating new ones (*Reserve VP / VC Subnetwork Connection* and *Activate VP / VC Subnetwork Connection*). The reconfiguration process is controlled by the I-PNO. Further details can be found in the Establish and Release MSC description.

List of Functions

- Establish and Release MSC functions.

7.2.5 Continuity Check MSC

The Continuity Check will be implemented in an optional way: the Continuity Check functions can only be used by the initiating PNO, with PNOs which are able to support it.

NOTE 1: If a PNO doesn't support Continuity Check functionality, but the PNOs extremity of the segment support it, the initiating PNO can use CC functions with the PNOs extremity of the segment.

Summary Description

The initiating PNO of a vpConnection is allowed to initiate (and stop) a Continuity Check over a vpConnection segment. This segment can be either the whole vpConnection or part of the vpConnection.

The Continuity Check may involve one or more PNOs.

The Initiating PNO requests to all the PNOs involved in the vpConnection segment to stop the use of the F4 flow over their managed VP Subnetwork Connection (*stopF4Flow-request*).

Then the Initiating PNO requests the creation of the pnoBidirectionalContinuityMonitor for the Continuity Check to the extremity PNOs of the segment (*Create bidirectionalContinuityMonitor for Continuity Check-request*). The addressed PNOs respond with positive or negative acknowledgement.

If case of positive response, the Initiating PNO requests the activation of the Sink function to the extremity PNOs of the segment (*Activate bidirectionalContinuityMonitor Sink for Continuity Check-request*).

NOTE 2: The Source function is activated at the bidirectionalContinuityMonitor creation.

When both extremities Sink and Source are activated, the Continuity Check is performed.

When the Continuity Check has to be stopped, the Initiating PNO first asks for the Sink deactivation to the extremity PNO of the segment (*Deactivate bidirectionalContinuityMonitor Sink for Continuity Check-request*). Then it asks for the bidirectionalContinuityMonitor deletion to the extremity PNOs of the segment (*Delete bidirectionalContinuityMonitor-request*).

At the end of the Continuity Check, the Initiating PNO allows the use of the F4 flow (*allowF4Flow-request*) over their managed VP Subnetwork Connection for all the PNOs involved in the vpConnection segment.

During the Continuity Check, if the extremity Sink detects a lossOfContinuity, the concerned PNO sends an indication to the Initiating PNO (*Report of Continuity Check*).

Operational State Changes on Bidirectional Continuity Monitor are notified to the Initiating PNO to inform it that the Continuity Check can’t be performed.

List of Functions

- Create Bidirectional Continuity Monitor for Continuity Check;
- Activate Bidirectional Continuity Monitor Sink for Continuity Check;
- Deactivate Bidirectional Continuity Monitor Sink for Continuity Check;
- Delete Bidirectional Continuity Monitor;
- Notification of Bidirectional Continuity Monitor Operational State change;
- Stop F4 flow;
- Allow F4 flow;
- Report of Continuity Check.

7.3 Management Functions (MF)

7.3.1 Activate Change

Description

This function requests the other PNOs involved to activate the connection with the modified parameters.

This message will be sent in parallel to all involved PNOs. The PNOs respond with the result of the activation.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection id	Identity assigned to the connection
Response:	Activate Change result	Result of the activate change attempt

7.3.2 Activate VP / VC Subnetwork Connection

Description

This function requests the other PNOs involved to activate the VP / VC connection between the specified input and output ports of their subnetwork, or between the input port and the user for the ***Z*** subnetwork. This message will be sent in parallel. The other PNOs respond with the result of the activation.

This MF is applied only if the connection is not unlocked automatically in accordance with the schedule.

This function can be used for performing continuity check in a given connection, outside of the agreed scheduling.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection id	Identity assigned to the connection
Response:	Activation result	Result of the activation attempt

7.3.3 Cancel Change

Description

If another PNO (Transit or ***Z***) answers with a negative result to the change reservation request, all the already confirmed changed reservations have to be cancelled by the NMS of the Initiating PNO.

This function requests the other PNOs involved to cancel the changes to the parameters of the VP / VC connection.

This can be done in parallel for each reservation of a subnetwork connection by sending a *Cancel Change-request* primitive.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
Response:	Cancel Change Result	Result of the cancel change attempt

7.3.4 Release VP / VC Subnetwork Connection

Description

If any PNO in the path (Transit or ***Z***) answers with a negative result to the reservation request, or if the Initiating PNO takes the initiative of releasing, all the already confirmed reservations have to be cancelled by the Initiating PNO. This can be done in parallel for each reservation of a subnetwork connection by sending a *Release VP / VC Subnetwork Connection-request primitive.*

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
Response:	Release Result	Result of the cancel reservation attempt

7.3.5 Change Reservation

Description

This function requests the other involved PNOs to check if they have resources available to support the altered parameters for the connection specified by the connection identifier.

Upon reception of this request a transit PNO should check the availability of the new scheduler between the linked input and output ports of its subnetwork; if it is an A or a Z PNO the check has to be done between the input port and the user.

The PNOs respond with the new scheduling parameters for the already established connection if the check has been successful, or with an error message if the check has shown no possibility to change the scheduler. The connection reservation will be kept anyway unchanged.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
	Forward traffic descriptor (new)	Altered peak cell rate from **A** to **Z**
	Backward traffic descriptor (new)	Altered peak cell rate from **Z** to **A**
	Schedule (new)	Changed Schedule
Response:	Change reservation result	Result of the change reservation attempt

7.3.6 Check Available Cell Rate

Description

When a new VP / VC connection establish request is received from the service layer, it is reasonable to check that the **Z** PNO has the necessary cell rate available on at least one of its incoming inter-PNO links, before expending network resources in establishing the connection. From a PNO's point of view it is only a check that the **Z** PNO has the cell rate available and not for the total amount of available cell rate.

The request can also be sent to a transit PNO. In this case the Agent-PNO will always consider itself as the Z side of the link with regard to the direction of the bandwidth in the request.

On receiving a VP / VC connection establish request from the Service Layer, the Initiating PNO can send out this message to the ***Z*** PNO or to a Transit-PNO. The message will include the peak cell rate required in both directions for the requested VP / VC connection. The **Z** PNO or the Transit-PNO (in the Initiating PNO’s view) will then check on which of its inter-PNO links this cell rate is available and respond to the Initiating PNO with the corresponding list of neighbouring PNOs.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
	Forward traffic descriptor	Peak cell rate from **A** to **Z**
	Backward traffic descriptor	Peak cell rate from **Z** to **A**

	PNO Subnetwork Id	Identity of Initiating PNO
	Schedule	List of activation times, dates and durations
Response:	Check Cell Rate Response	List of inter-PNO links which can accommodate the requested cell rate

7.3.7 Deactivate VP / VC Subnetwork Connection

Description

In any case of deactivation of a subnetwork connection, this primitive is sent to each involved PNO. It can be done in parallel.

This function requests the other PNOs involved to deactivate the connection between the specified input and output ports of their subnetwork or between the input port and the user for the ***Z*** subnetwork. The other PNOs respond with the result of the deactivation.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
Response:	Deactivation Result	Result of the deactivation attempt

7.3.8 Destination User Checking

Description

When a new VP / VC connection establish request is received from the Service Layer, it is reasonable to verify that the ***Z*** user is able or wants to support the requested connection before expending network resources in establishing the connection. From a PNO's point of view only a check that the ***Z*** PNO can accept the connection is performed.

On receiving a VP / VC connection establish request from the Service Layer, the Initiating PNO will send out a *Destination User Checking-request* message to the ***Z*** PNO, and will include the peak cell rate (in both directions), and the ***A*** and ***Z*** addresses for the requested VP / VC Connection. The ***Z*** PNO will then perform the ***Z*** user checking and respond with the result to the Initiating PNO.

In case the ***Z*** user is under the ***Z*** PNO domain (star approach), the ***Z*** PNO will perform the ***Z*** user checking and respond with the result to the Initiating PNO.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
	A Address	E.164 address of the **A** user
	Z Address	E.164 address of **Z** user
	Forward Traffic Descriptor	Peak cell rate from **A** to **Z**
	Backward Traffic Descriptor	Peak cell rate from **Z** to **A**
	Schedule	List of activation times, dates and durations
	PNO Subnetwork Id	Identity of the Initiating PNO

Response:	Check Response	Result of **Z** user check
	Check Reason	Reason for failure of check
	PNO Subnetwork Id	Identity of the reporting PNO

7.3.9 Reserve VP / VC Subnetwork Connection

Description

This function requests the other involved PNOs to reserve a path between the specified input and output points of their subnetwork, or between the input port and the user, in case of the A and the **Z** subnetwork. The other PNOs respond with the result of the reservation.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
	Subnetwork Connection Id (nearend)	Identification of the nearend Subnetwork connection
	PNO Subnetwork Id (far-end)	Identification of the far-end PNO Subnetwork
	A Address	E.164 address of **A** user
	Z Address	E.164 address of **Z** user
	Z VP / VCI (optional)	VP / VCI to be used by the **Z** PNO only to allow user transparency of reconfiguration
	Forward traffic descriptor	Peak cell rate from **A** to **Z**
	Backward traffic descriptor	Peak cell rate from **Z** to **A**
	Forward QoS class	An indication of the class of QoS
	Backward QoS class	An indication of the class of QoS
	Schedule	List of activation times, dates and durations
Response:	Subnetwork Connection Id (far-end)	Identification of the far-end Subnetwork Connection
	Reservation result	Result of the reservation attempt

7.3.10 Topology Info Changes

Description

In order that each PNO is able to determine appropriate routes for VP / VC connections (both when establishing or reconfiguring VP / VC connections) it is essential that each PNO keeps an up to date map of the topology of the network. Using the co-operative management approach, it is only necessary for the PNOs to know the topology of the inter-PNO connections, not the topology of each PNO's own network. These inter-PNO connections will usually be the international links except in some countries where there are multiple PNOs.

To be able to keep the local inter-PNO topology map up to date, each PNO shall send an indication of any changes that it makes to its inter-PNO transmission links to all the other PNOs. A similar message can be sent by a new PNO when joining the network. The details that should be sent shall include the installed capacity of the transmission link, the identity of the PNOs at each end and the status of the transmission links.

When two PNOs agree to connect together a new transmission link, it is assumed that there will be an agreement as to which one of the PNOs will send out the topology change and assign an identity to the link.

This function has two elements that require a different set of parameters, one for handling the addition or removal of PNOs and one for handling the addition, modification and removal of inter PNO links.

For PNO addition/removal

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	PNO Subnetwork Id	Identity assigned to the PNO
	Message Status	Addition or removal of PNO
Response:	PNO Subnetwork Id	Identity of PNO sending confirmation

For Inter PNO link addition/modification/removal

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	ATM Path Cell Rate	Capacity of the Inter-PNO link
	Own PNO Subnetwork Id	Identity of the own PNO
	Neighbour PNO Subnetwork Id	Identity of the neighbour PNO
	Message Type	Add, modify or remove inter-PNO link
	Link Status	Link active or faulty
	Inter PNO ATM Path QoS	QoS of physical link
	Inter PNO ATM Path Id	Identity of the link
Response:	PNO Subnetwork Id	Identity of PNO sending confirmation
	Inter PNO ATM Path Id	Identity of the link

7.3.11 Cancel VP / VC Subnetwork Connection

This function allows a zPNO to notify the initiating PNO that it wants the initiator to release, for a particular reason, a subnetwork connection in the zPNO's net.

It is a non confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	VP / VC Connection Id	Identity assigned to the connection
	disconnect Cause	The zPNO's reason for his request
Response:	None	

7.3.12 Activate Bidirectional Continuity Monitor Sink for Continuity Check

Description

This function requests to the extremity PNOs of the segment involved in a segment Continuity Check to activate the Sink mechanism for OAM flow.

NOTE: The Source mechanism is activated at bidirectionalContinuityMonitor Creation.

The message associated with this function will be sent by the Initiating PNO to the extremity PNOs of the segment. The extremity PNOs of the segment respond with the result of the activation.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	Bidirectional continuity monitor Id	Identity of the bidirectional continuity monitor to activate
	ControlContinuity CheckInformation	Activate Sink
Response:	Activation result	Result of the activation attempt

7.3.13 Allow F4 flow

Description

This function allows a PNO to use the F4 flow over a Subnetwork Connection or an inter-PNO link.

The message associated with this function is sent by the Initiating PNO to all the PNOs involved in a segment Continuity Check when this check is ended.

It is confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	PNO VP Subnetwork Connection Id	Identity assigned to the concerned PNO Subnetwork Connection
	Allow test	
Response:	Activation result	Result of the activation attempt

7.3.14 Create Bidirectional Continuity Monitor for Continuity Check

Description

A continuity check OAM flow is used in a vpConnection in order to monitor the continuity of this vpConnection.

The Initiating PNO creates a bidirectionalContinuityMonitor for Continuity Check at the entry edge or exit edge of the extremity PNOs of the segment.

For that creation, the Initiating PNO sends a Create bidirectionalContinuityMonitor message to the extremity PNOs of the segment.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	Bidirectional continuity monitor Id	Identity assigned to the created bidirectional continuity monitor
	FlowDirection	Direction in/from which the OAM flow is generated/received
Response:	Creation result	Result of the create

The bidirectionalContinuityMonitor Source mechanism represents the source of a segment OAM flow. The bidirectionalContinuityMonitor Source mechanism is activated at object creation.

7.3.15 Deactivate Bidirectional Continuity Monitor Sink for Continuity Check

Description

This function requests to the extremity PNOs of the segment involved in a segment Continuity Check to deactivate the Source/Sink mechanism for OAM flow.

The function is used first to deactivate the Sink mechanism at each extremity. The Source mechanism will be deleted at object deletion.

The message associated with this function will be sent by the Initiating PNO to the extremity PNOs of the segment. The extremity PNOs of the segment respond with the result of the deactivation.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	Bidirectional continuity monitor Id	Identity of the bidirectional continuity monitor to deactivate
	ControlContinuity CheckInformation	Deactivate Sink
Response:	Deactivation result	Result of the deactivation attempt

7.3.16 Delete Bidirectional Continuity Monitor

Description

The Initiating PNO deletes bidirectionalContinuityMonitor objects created for Continuity Check at the entry edge or exit edge of the extremity PNOs of the segment.

For that deletion, the Initiating PNO sends a Delete bidirectionalContinuityMonitor message to the extremity PNO of the checked segment of the connection after the end of the check.

It is a confirmed type of operation. The following parameters are associated with the Request and Response primitives of the function:

Request:	Bidirectional continuity monitor Id	Identity assigned to the deleted bidirectional continuity monitor
Response:	Deletion result	Result of the delete

7.3.17 Notification of Bidirectional Continuity Monitor Operational State Change

Description

When an Operational State change occurs on a pnoBidirectionalContinuityMonitor, the Initiating PNO of the vpConnection must be informed that the Continuity Check can’t be performed or must be stopped if it is already started.

This Operational State change will be sent using the normal stateChange Notification.

7.3.18 Report of Continuity Check

Description

The bidirectional continuity monitor receives a continuity check OAM flow in a vpConnection.

When the bidirectional continuity monitor Sink part detects a disruption on the received flow, a Report of Continuity Check Notification is sent to the Initiating PNO.

It is a non-confirmed type of operation. The following parameters are associated with the Request primitives of the function:

Request:	Bidirectional continuity monitor Id	Identity assigned to the bidirectional continuity monitor which has detected the connection disruption
	Probable Cause	Set to LOC: loss of continuity (17)

7.3.19 Stop F4 flow

Description

This function inhibits a PNO from using the F4 flow over a Subnetwork Connection or an inter-PNO link.

The message associated with this function is sent by the Initiating PNO to all the PNOs involved in a segment Continuity Check when before this check is performed.

It is a confirmed type of operation. The following parameters are associated with the Request and response primitives of the function:

Request:	PNO VP Subnetwork Connection Id	Identity assigned to the concerned PNO Subnetwork Connection
	Inhibit test	
Response:	Deactivation result	Result of the deactivation attempt

8 Management information model

8.1 Introduction

NOTE 1: The visibility across the X-interface of object classes and their attributes, as well as access control information are a subject for agreements between Operators.

NOTE 2: The present model is restricted to the VP / VC Bearer Service.

The information model described in this subclause is inherited from the ones contained in I-ETS 300 653 [3] and ITU-T Recommendation I.751 [5].

For the specialization of the X Managed Object Classes the concept of Profile, as defined in I-ETS 300 653 [3], is used. According to this definition, to profile a managed object is to add additional normative text which restricts conditionality (e.g. specifies that a conditional package is or is not present) and adds behaviour to it.

Using this concept, the description of the Managed Object Classes is done in two steps (see figure 8).

1) A profile of the object classes of I-ETS 300 653 [3] and ITU-T Recommendation I.751 [5] adapted for the X is defined;

2) The X Managed Object Classes are inherited from the profile defined in step 1.

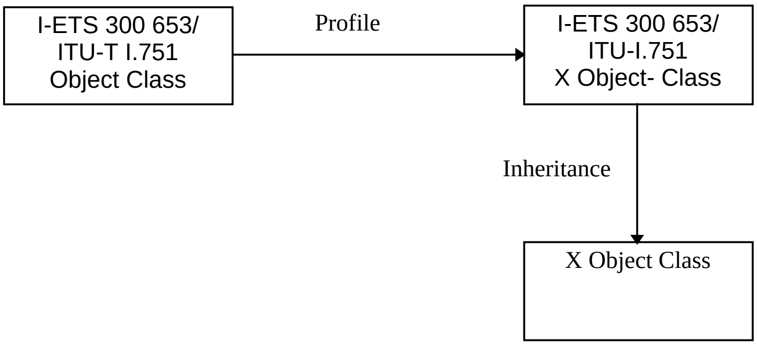
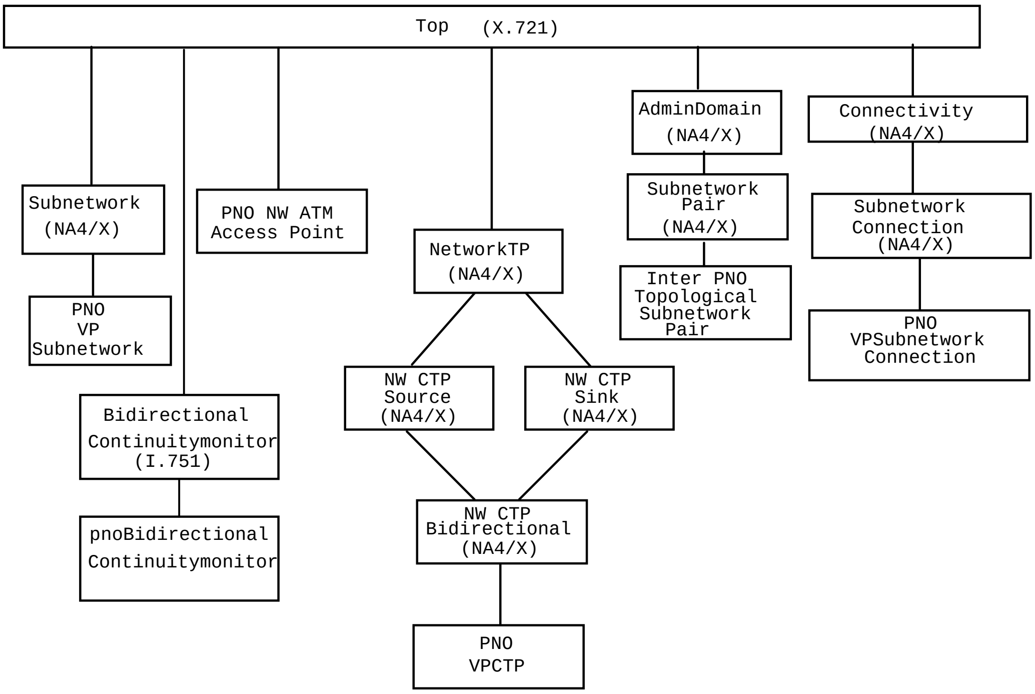


Figure 8: Methodology for the X Object Classes specialization

8.2 Inheritance tree



X.721: ITU-T Recommendation X.721 [2]-Definition of Management Information
NA4: I-ETS 300 653 [3] - Network Level View Managed Object Classes
NA4/X: Profiled object classes from I-ETS 300 653 [3]

NOTE: For reasons of simplicity, only the objects defined in this specification are shown in the inheritance tree.

Figure 9: Inheritance tree

8.3 Entity relationship diagram

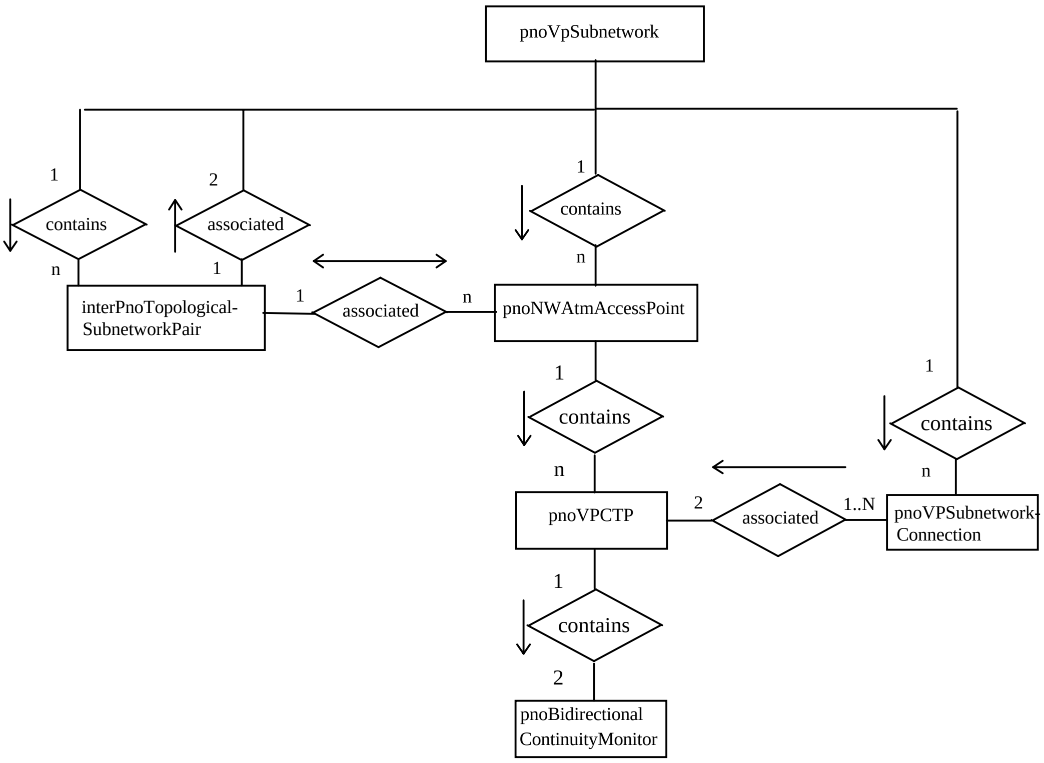


Figure 10: Entity relationship diagram

8.4 I-ETS 300 653 and ITU-T Recommendation I.751 Object Classes adapted for the X

8.4.1 Connectivity

PROFILE NOTE:

The attribute *signalid* will not be used in the X Managed Object Class. Its value should be set to NULL.

The conditional packages

- assignmentStatePackage;
- availabilityStatusPackage;
- lifecycleStatePackage;
- alarmSeverityAssignmentPointerPackage;
- supportedByPackage;
- userLabelPackage; and
- qualityOfConnectivityServicePackage;

are not required for the X Managed Object Classes.

The conditional packages:

- createDeleteNotificationsPackage;

- attributeValueChangeNotificationPackage;
- stateChangeNotificationPackage;
- administrativeStatePackage;
- operationalStatePackage;
- tmnCommunicationAlarmInformationPackage; and
- zEndNWTPListPackage;

are mandatory for the X Managed Classes.

```
connectivity MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Recommendation X.721 [2]/ISO/IEC 10165-2:92":top;
CHARACTERIZED BY
connectivityPackage PACKAGE
BEHAVIOUR
connectivityBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
ATTRIBUTES
signalid GET,
mode GET,
aEndNWTPList GET,
"ITU-T Recommendation M.3100 [1]:92":directionality GET;

CONDITIONAL PACKAGES
"ITU-T Recommendation M.3100 [1]:92":createDeleteNotificationsPackage PRESENT IF " See
I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":attributeValueChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":stateChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
administrativeStatePackage PRESENT IF " See I-ETS 300 653 [3]",
assignmentStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation X.721 [2]": availabilityStatusPackage PRESENT IF " See I-ETS 300 653 [3]",
lifecycleStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":operationalStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":tmnCommunicationsAlarmInformationPackage PRESENT IF " See
I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":alarmSeverityAssignmentPointerPackage PRESENT IF " See
I-ETS 300 653 [3]",
supportedByPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":userLabelPackage PRESENT IF " See I-ETS 300 653 [3]",
qualityOfConnectivityServicePackage PRESENT IF " See I-ETS 300 653 [3]",
zEndNWTPListPackage PRESENT IF " See I-ETS 300 653 [3]";
;;
REGISTERED AS {iets300 653 [3]MObjectClass 6};
```

8.4.2 Subnetwork Connection

PROFILE NOTE:

The conditional packages:

- compositePointerPackage;
- componentPointerPackage;
- userLabelPackage;
- durationSchedulingPackage;
- dailyBasisSchedulingPackage;
- weeklyBasisSchedulingPackage;
- monthlyBasisSchedulingPackage;
- occasionalSchedulingPackage;

are not required for the X Managed Object Classes.

```
subNetworkConnection MANAGED OBJECT CLASS

DERIVED FROM connectivity;
CHARACTERIZED BY
subNetworkConnectionPackage PACKAGE
BEHAVIOUR
subNetworkConnectionBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
ATTRIBUTES
subnetworkConnectionId GET;

CONDITIONAL PACKAGES
compositePointerPackage PRESENT IF " See I-ETS 300 653 [3]",
componentPointerPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":userLabelPackage PRESENT IF " See I-ETS 300 653 [3]",
durationSchedulingPackage PRESENT IF " See I-ETS 300 653 [3]",
dailyBasisSchedulingPackage PRESENT IF " See I-ETS 300 653 [3]",
weeklyBasisSchedulingPackage PRESENT IF " See I-ETS 300 653 [3]",
monthlyBasisSchedulingPackage PRESENT IF " See I-ETS 300 653 [3]",
occasionalSchedulingPackage PRESENT IF " See I-ETS 300 653 [3]";

REGISTERED AS {iets300 653 [3]MObjectClass 24};
```

8.4.3 Network TP

PROFILE NOTE:

The attribute *signalid* will not be used in the X Managed Object Class. Its value should be set to NULL.

The attribute "mode" is currently not used but will be kept. It’s value will always be pointToPoint(0).

The conditional packages connectivityPointerPackage, neAssignmentPackage, tmnCommunicationsAlarmInformationPackage, sncPointerPackage, networkTPPointerPackage, userLabelPackage, assignmentStatePackage, availabilityStatusPackage, lifecycleStatePackage, supportedByPackage, attributeValueChangeNotificationPackage, administrativeStatePackage, operationalStatePackage and stateChangeNotificationPackage are not required for the X Managed Object Class.

```
networkTP MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Recommendation X.721 [2]/ISO/IEC 10165-2:92":top;
CHARACTERIZED BY
"ITU-T Recommendation M.3100 [1]:92":createDeleteNotificationsPackage,
networkTPPackage PACKAGE
BEHAVIOUR
networkTPBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
ATTRIBUTES
signalid GET,
mode GET;

CONDITIONAL PACKAGES
connectivityPointerPackage PRESENT IF " See I-ETS 300 653 [3]",
neAssignmentPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":tmnCommunicationsAlarmInformationPackage PRESENT IF " See
I-ETS 300 653 [3]",
sncPointerPackage PRESENT IF " See I-ETS 300 653 [3]",
networkTPPointerPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":attributeValueChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":userLabelPackage PRESENT IF " See I-ETS 300 653 [3]",
administrativeStatePackage PRESENT IF " See I-ETS 300 653 [3]",
assignmentStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation X.721 [2]":availabilityStatusPackage PRESENT IF " See I-ETS 300 653 [3]",
lifecycleStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":operationalStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":stateChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
supportedByPackage PRESENT IF " See I-ETS 300 653 [3]";
;;
REGISTERED AS {iets300 653 [3]MObjectClass 18};
```

8.4.4 Network CTP Sink

PROFILE NOTE:

The conditional packages channelNumberPackage, ctpInstancePackage, networkCTPPackage and serverTTPPointerPackage, are not required for the X Managed Object Class.

```
networkCTPSink MANAGED OBJECT CLASS

DERIVED FROM networkTP;
CHARACTERIZED BY
networkCTPSinkPackage PACKAGE
BEHAVIOUR
networkCTPSinkBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
CONDITIONAL PACKAGES
"ITU-T Recommendation M.3100 [1]:92":channelNumberPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":ctpInstancePackage PRESENT IF " See I-ETS 300 653 [3]",
networkCTPPackage PRESENT IF " See I-ETS 300 653 [3]",
serverTTPPointerPackage PRESENT IF " See I-ETS 300 653 [3]";
;;
REGISTERED AS {iets300 653 [3]MObjectClass 15};
```

8.4.5 Network CTP Source

PROFILE NOTE:

The conditional packages channelNumberPackage, ctpInstancePackage, networkCTPPackage and serverTTPPointerPackage, are not required for the X Managed Object Class.

```
networkCTPSource MANAGED OBJECT CLASS

DERIVED FROM networkTP;
CHARACTERIZED BY
networkCTPSourcePackage PACKAGE
BEHAVIOUR
networkCTPSourceBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
CONDITIONAL PACKAGES
"ITU-T Recommendation M.3100 [1]:92":channelNumberPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":ctpInstancePackage PRESENT IF " See I-ETS 300 653 [3]",
networkCTPPackage PRESENT IF " See I-ETS 300 653 [3]",
serverTTPPointerPackage PRESENT IF " See I-ETS 300 653 [3]";
;;
REGISTERED AS {iets300 653 [3]MObjectClass 16};
```

8.4.6 Network CTP Bi-directional

```
networkCTPBidirectional MANAGED OBJECT CLASS

DERIVED FROM
networkCTPSink,
networkCTPSource;

REGISTERED AS {iets300 653 [3]MObjectClass 14};
```

8.4.7 Admin Domain

PROFILE NOTE:

The conditional packages adminDomainIdPackage, systemTitlePackage and userLabelPackage, are not required for the X Managed Object Class.

```
adminDomain MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Recommendation X.721 [2]/ISO/IEC 10165-2:92":top;
CHARACTERIZED BY
```

```
adminDomainPackage PACKAGE
BEHAVIOUR
adminDomainBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
;
CONDITIONAL PACKAGES
adminDomainIdPackage PRESENT IF " See I-ETS 300 653 [3]",
systemTitlePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":userLabelPackage PRESENT IF " See I-ETS 300 653 [3]";
;;
REGISTERED AS {iets300 653 [3]MObjectClass 2};
```

8.4.8 Subnetwork Pair

PROFILE NOTE:

The attributes *trailList* and *signalid* will not be used in the X-interface Managed Object Class. The attribute *trailList* will always empty. The attribute *signalid* will be set to NULL.

```
subnetworkPair MANAGED OBJECT CLASS

DERIVED FROM adminDomain;
CHARACTERIZED BY
subnetworkPairPackage PACKAGE
BEHAVIOUR
networkTPBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
ATTRIBUTES
aEndPoint GET,
zEndPoint GET,
trailList GET,
subNetworkPairId GET,
signalid GET;
;;
REGISTERED AS {iets300 653 [3]MObjectClass 25};
```

8.4.9 Sub-Network

PROFILE NOTE:

The conditional packages attributeValueChangeNotificationPackage, signalidPackage, userLabelPackage, assignmentStatePackage, availabilityStatusPackage, lifecycleStatePackage, supportedByPackage, containedNWCTPListPackage, containedNWTTPListPackage, containedLinkListPackage, containedSubNetworkListPackage, containedInSubNetworkListPackage and linkPointerListPackage are not required for the X Managed Object Class.

The conditional packages stateChangeNotificationPackage, subNetworkIdPackage, administrativeStatePackage and operationalStatePackage, are mandatory for the X Managed Class.

```
subNetwork MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Recommendation X.721 [2]/ISO/IEC 10165-2:92":top;;
CHARACTERIZED BY
"ITU-T Recommendation M.3100 [1]:92":createDeleteNotificationsPackage,
subNetworkPackage PACKAGE
BEHAVIOUR
subNetworkBehaviour BEHAVIOUR
DEFINED AS
" See I-ETS 300 653 [3]"
;;
CONDITIONAL PACKAGES
"ITU-T Recommendation M.3100 [1]:92":stateChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":attributeValueChangeNotificationPackage PRESENT IF " See
I-ETS 300 653 [3]",
signalidPackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":userLabelPackage PRESENT IF " See I-ETS 300 653 [3]",
subNetworkIdPackage PRESENT IF " See I-ETS 300 653 [3]",
administrativeStatePackage PRESENT IF " See I-ETS 300 653 [3]",
assignmentStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation X.721 [2]":availabilityStatusPackage PRESENT IF " See I-ETS 300 653 [3]",
lifecycleStatePackage PRESENT IF " See I-ETS 300 653 [3]",
"ITU-T Recommendation M.3100 [1]:92":operationalStatePackage PRESENT IF " See I-ETS 300 653 [3]",
supportedByPackage PRESENT IF " See I-ETS 300 653 [3]",
containedNWCTPListPackage PRESENT IF " See I-ETS 300 653 [3]",
containedNWTTPListPackage PRESENT IF " See I-ETS 300 653 [3]",
containedLinkListPackage PRESENT IF " See I-ETS 300 653 [3]",
containedSubNetworkListPackage PRESENT IF " See I-ETS 300 653 [3]",
containedInSubNetworkListPackage PRESENT IF " See I-ETS 300 653 [3]",
linkPointerListPackage PRESENT IF " See I-ETS 300 653 [3]";
REGISTERED AS {iets300 653 [3]MObjectClass 23};
```

8.4.10 bidirectionalContinuityMonitor

The conditional package flowDirectionPackage is mandatory in the X Managed Object Class.

```
bidirectionalContinuityMonitor MANAGED OBJECT CLASS

    DERIVED FROM "Rec. X.721|ISO/IEC-10165-2":top;
    CHARACTERIZED BY
    "ITU-T M.3100":tmnCommunicationsAlarmInformationPackage,
    "ITU-T M.3100":stateChangeNotificationPackage,
    bidirectionalContinuityMonitorPackage PACKAGE
 BEHAVIOUR bidirectionalContinuityMonitorBeh;
 ATTRIBUTES
    continuityMonitorId         GET,
sinkCCMechanismActive       GET,
sourceCCMechanismActive     GET,
"Rec. X.721|ISO/IEC-10165-2":operationalState       GET;

ACTIONS
controlCC;;;

CONDITIONAL PACKAGES
flowDirectionPackage
PRESENT IF "the monitor object instance is contained in CTP";
REGISTERED AS { i751ObjectClass 7 };

bidirectionalContinuityMonitorBeh BEHAVIOUR
  DEFINED AS
"This managed object models a Continuity Check (CC) OAM flow. If this object is named by a
vpCTPBidirectional or vcCTPBidirectional object, then it represents the termination of a segment OAM
flow.  If this object is named by a vpTTPBidirectional or vcTTPBidirectional object, then it
represents the termination of an end-to-end OAM flow.

The controlCC action may be used to manage the activation and deactivation of the continuity check
OAM cell mechanism. The controlCC action allows to separately activate the sink and source
mechanisms in the bidirectionalContinuityMonitor object.  The activation of the sink mechanisms will
be reflected by setting the values of the sinkCCMechanismActive attribute to TRUE.  Similarly, the
deactivation of the source mechanism will be reflected by setting the sourceCCMechanismActive to
FALSE.

When the sinkCCMechanismActive attribute is TRUE and the VP/VC is detected as disrupted, a
communications alarm with the probableCause set to lossOfContinuity (LOC) is notified.

This object is  created and deleted by the managing system, using the CMIS M-CREATE and M-DELETE
services .

These  objects  shall  be  automatically  deleted  when  the  containing  instance  of  the
vpCTPBidirectional  and  vcCTPBidirectional  object  class  is  deleted.

The managing system can request the creation of the bidirectionalContinuityMonitor object instance
and the activation of the Continuity Check function at the same time in the CREATE request by
setting the sourceCCMechanismActive attribute (and possibly the sinkCCMechanismActive attribute) to
TRUE.
When the managing system requests the creation of a bidirectionalContinuityMonitor object instance
without the activation of Continuity Check function, the managed system shall set the
sourceCCMechanismActive attribute to FALSE.

The managing system can request the creation of the bidirectionalContinuityMonitor object instance
and the activation of the Continuity Check function at the same time in the CREATE request by
setting the sinkCCMechanismActive attribute (and possibly the sourceCCMechanismActive attribute) to
TRUE.
When the managing system requests the creation of a bidirectionalContinutyMonitor object instance
without the activation of the Continuity Check function, the managed system shall set the
sinkCCMechanismActive attribute to FALSE.";
```

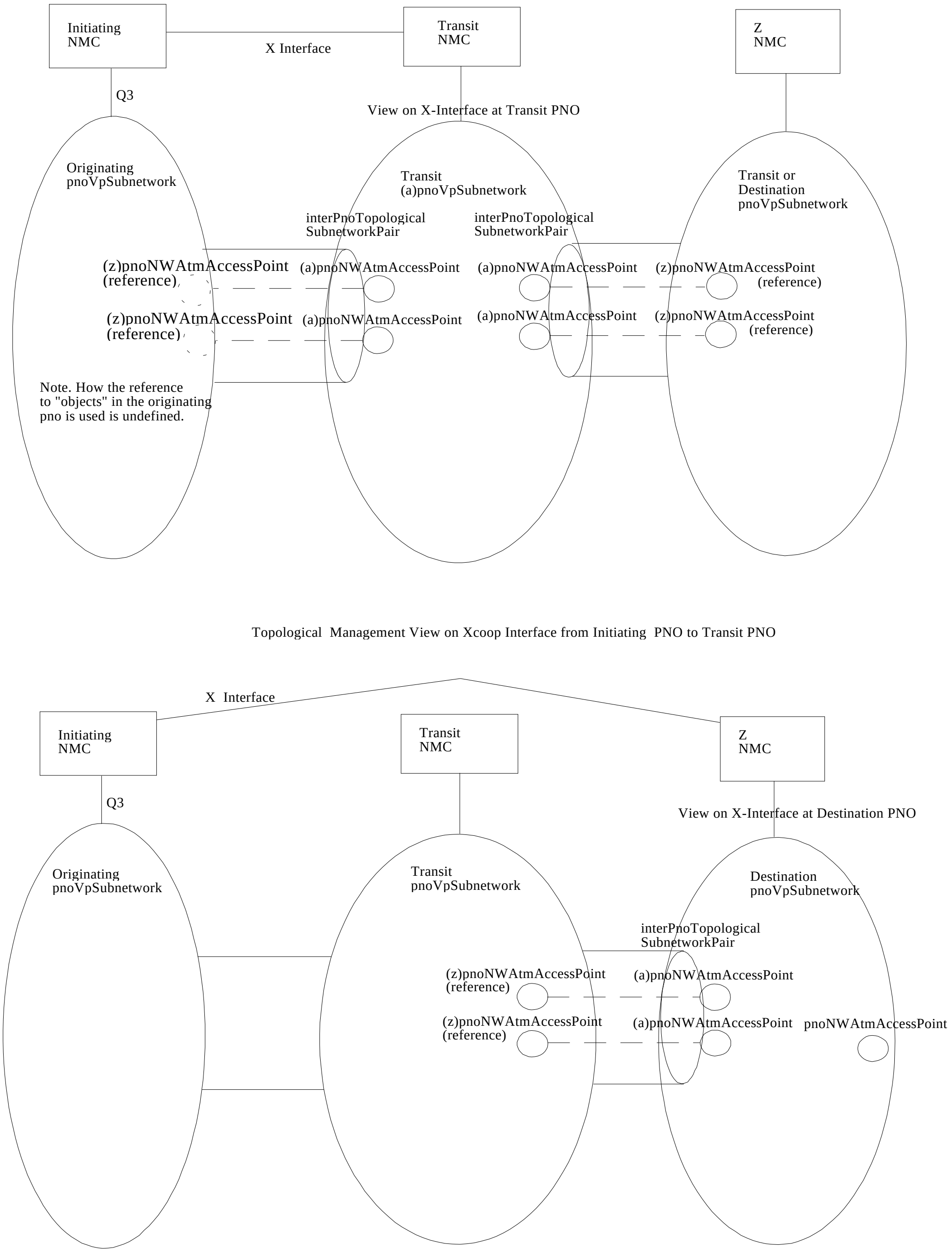
8.5 ATM VP X Object Classes

8.5.1 Introduction

The Information Model described represents the view of the ***Initiating*** PNO on the Transit and ***Z*** PNOs.

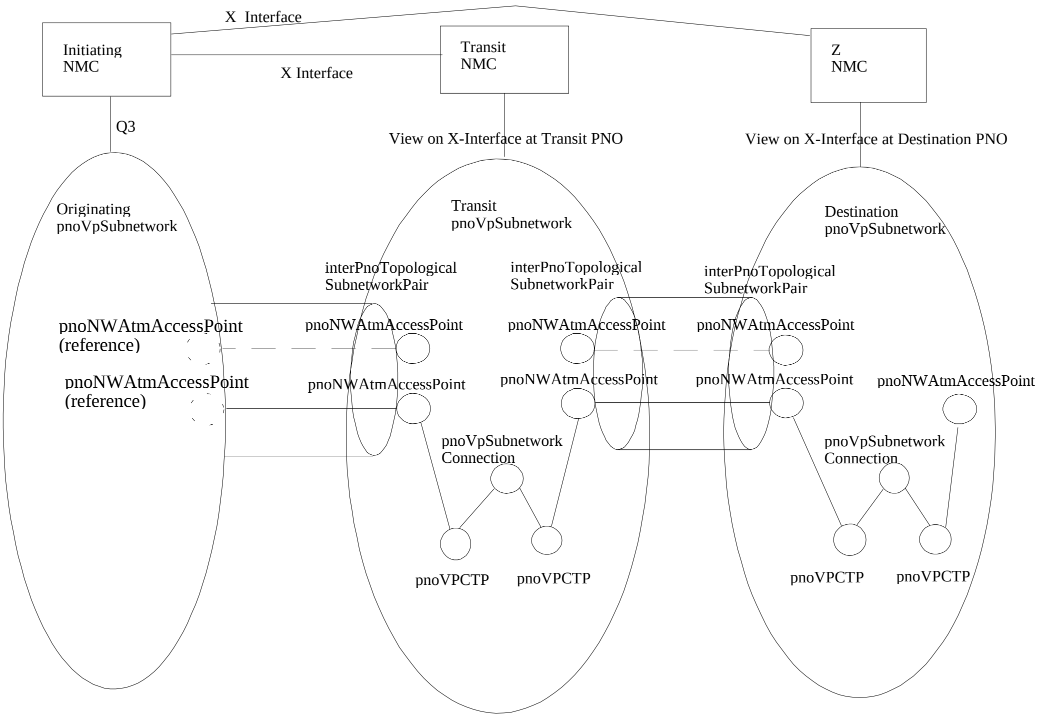
The ***Initiating*** PNO takes the role of Manager for the establishment and control of a VP Connection. The created instances of the Objects are only managed by the ***Initiating*** PNO.

The following figures illustrate the management view from the ***Initiating*** PNO. Figure 11 represents the management view on the topological objects. Figure 12 represents the view after a VP Connection has been setup. In these examples the Initiating PNO and the A PNO are the same.


Initiating NMC
X Interface
Transit NMC
Z NMC
Q3
View on X-Interface at Transit PNO
Originating pnoVpSubnetwork
Transit (a)pnoVpSubnetwork
Transit or Destination pnoVpSubnetwork
interPnoTopological SubnetworkPair
interPnoTopological SubnetworkPair
(z)pnoNWAtmAccessPoint (reference)
(a)pnoNWAtmAccessPoint
(a)pnoNWAtmAccessPoint
(z)pnoNWAtmAccessPoint (reference)
(z)pnoNWAtmAccessPoint (reference)
(a)pnoNWAtmAccessPoint
(a)pnoNWAtmAccessPoint
(z)pnoNWAtmAccessPoint (reference)
Note. How the reference to "objects" in the originating pno is used is undefined.
Topological Management View on Xcoop Interface from Initiating PNO to Transit PNO
X Interface
Initiating NMC
Transit NMC
Z NMC
Q3
View on X-Interface at Destination PNO
Originating pnoVpSubnetwork
Transit pnoVpSubnetwork
Destination pnoVpSubnetwork
interPnoTopological SubnetworkPair
(z)pnoNWAtmAccessPoint (reference)
(a)pnoNWAtmAccessPoint
(z)pnoNWAtmAccessPoint (reference)
(a)pnoNWAtmAccessPoint
pnoNWAtmAccessPoint

Topological Management View on X coop Interface from Initiating PNO to Z PNO

Figure 11: Topological Management View on the Xcoop Interface


X Interface
Initiating NMC
X Interface
Transit NMC
Z NMC
Q3
View on X-Interface at Transit PNO
View on X-Interface at Destination PNO
Originating pnoVpSubnetwork
Transit pnoVpSubnetwork
Destination pnoVpSubnetwork
interPnoTopological SubnetworkPair
interPnoTopological SubnetworkPair
interPnoTopological SubnetworkPair
pnoNWAtmAccessPoint (reference)
pnoNWAtmAccessPoint (reference)
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoNWAtmAccessPoint
pnoVpSubnetwork Connection
pnoVpSubnetwork Connection
pnoVPCTP
pnoVPCTP
pnoVPCTP
pnoVPCTP

Configuration Of Managed Objects on X Interface After A VP Connection is Made

Figure 12: Management View after a VP Connection has been setup

For describing the topology in the Information Model, the A and Z Subnetworks and Access Points correspond to the name of the PNOs taken in the alphabetical order, according to what is specified in the ITU-T Recommendation M.1400 [4]. In the context of a VP Connection, this notation is used independently for naming the Origin (A) and Destination (Z).

The VP Connection Id that is used in subclause 8.3 for identifying uniquely the end-to-end VP Connection, is composed in the Information Model of two attributes:

- InitiatingPnoSubnetwork - Which identifies the Initiating Subnetwork;

- InitiatingVpConnectionId - Which is an identifier of the VP Connection allocated by the Initiating Subnetwork.

8.5.2 VP Connection Fragment

PNO VP Subnetwork Connection

```
pnoVpSubnetworkConnection MANAGED OBJECT CLASS

DERIVED FROM "I-ETS 300 653 [3]":subNetworkConnection;
CHARACTERIZED BY

pnoVpSubnetworkConnectionPackage PACKAGE
BEHAVIOUR
pnoVpSubnetworkConnectionBehaviour BEHAVIOUR
DEFINED AS
"It represents a VP connection across a PNO subnetwork. The Initiating PNO only views this
connection as a whole, with no details regarding the identification of VPCCs which compose this
connection inside the PNO domain.
The attributes initiatingPnoSubnetworkId and intiatingVpConnectionId are used by the Initiating PNO
to assign an identity to the VP Connection.
The procedure Change Reservation is performed by setting the value of the attribute
changeReservationInformation to the new value.
The procedure Cancel Change is done by replacing the value of this attribute with the default value,
which is null.
```

```
The operationalState has two possible values: Disabled or Enabled. Disabled means that a failure has
occurred in the PNO VP Subnetwork Connection. Enabled means that the VP Subnetwork Connection is in
service.
The administrativeState has two possible values: Locked and unlocked. When an instance of this
object is created, the administrativeState shall be set to locked.
A stateChangeNotification (old value = locked, new value = unlocked) will be sent by each T/Z PNO to
the initiating PNO at the beginning of an activation slot.
A stateChangeNotification (old value = unlocked, new value = locked) will be sent to the initiating
PNO at the end of an activation slot.
The procedure Activate VP Subnetwork Connection is performed by setting the administrative state to
unlocked. The procedure Deactivate VP Subnetwork Connection is performed by setting the
administrative state to locked. The activation procedure is only used if the connection is not
unlocked automatically according to the schedule, and it can also be used to allow testing outside
of the scheduled timeslots. Whenever the operationalState has the value Disabled, the activation
procedures will not succeed, and the value of the administrativeState remains locked. If the
activation procedures are used outside of the scheduled timeslots, it may be refused, for whatever
reason, in which case the value of the administrativeState also remains locked.
When an instance of pnoVpSubnetworkConnection is created this either leads to the creation of one or
two instances of pnoVPCTP, or to the usage of instances of pnoVPCTP that are already used by other
pnoVpSubnetworkConnections. In the last case there may not be overlap with the time schedule of
these Connections.
The aEndNWTPList and the zEndNWTPList point to the pnoVPCTP instances associated with the
pnoVpSubnetworkConnection.
The notifications issued by this object are sent to the Initiating PNO.
The object Creation Notification should be sent after the Succesful Reserve Result.
The notification cancelVpNetworkConnectionNotification is sent by the Z PNO to the initiating PNO"
;;
ATTRIBUTES

initiatingPnoSubnetworkId GET,
intiatingPnoVpSubnetworkConnectionId GET,
forwardQoSClass GET,
backwardQoSClass GET,
vpSchedulers GET,

changeReservationInformation REPLACE-WITH-DEFAULT
                    DEFAULT VALUE                  ASN1XatmModule.defaultChangeReservationInfo
                    GET-REPLACE;
ACTIONS
activateChange;

CONDITIONAL PACKAGES
CancelVpNetworkConnectionNotificationPkg
PRESENT IF "it is a Z PNO Subnetwork for the corresponding VpConnection",
vpTestStatePackage
PRESENT IF "the T/Z PNO wants to know when he can make an internal CC"


REGISTERED AS {xatmObjectClass 1};
```

8.5.3 VP Connection Termination Point Fragment

PNO VPCTP

```
pnoVPCTP MANAGED OBJECT CLASS

DERIVED FROM "I-ETS 300 653 [3]":networkCTPbidirectional;
CHARACTERIZED BY
pnoVPCTPPackage PACKAGE
BEHAVIOUR
pnoVPCTPBehaviour BEHAVIOUR
DEFINED AS
" This managed object class represents the endpoint of a PNO VP Subnetwork Connection.
Two instances of the pnoVPCTP object are assigned to a pnoVpSubnetworkConnection instance.
The assignment of the VPI associated with this connection is done by the NearEnd NMS.
Contention for a particular VPI may be a problem i.e. the attempt to simultaneously allocate the
same VPI by two adjacent PNOs on the same physical link. To avoid this, a scheme should be adopted
whereby one PNO starts selecting VPIs from bottom end of the VPI range, and the other starts from
the top. This will be effected by agreement between PNOs for each particular physical link. The
Create/Delete Notifications are not used.
"
;;
ATTRIBUTES
"ITU-T Recommendation I.751":vpCTPId GET;
;;

REGISTERED AS {xatmObjectClass 2};
```

8.5.4 Topology Fragment

Inter PNO Topological Subnetwork Pair

```
interPnoTopologicalSubnetworkPair MANAGED OBJECT CLASS

DERIVED FROM "DTS 43316":subnetworkPair;
CHARACTERIZED BY
"ITU-T Recommendation M.3100 [1]:92":createDeleteNotificationsPackage,
"ITU-T Recommendation M.3100 [1]:92":stateChangeNotificationPackage,
"ITU-T Recommendation M.3100 [1]:92":attributeValueChangeNotificationPackage,
interPnoTopologicalSubnetworkPairPackage PACKAGE
BEHAVIOUR interPnoTopologicalSubnetworkPairDefinition BEHAVIOUR
DEFINED AS
"An interPnoTopologicalSubnetworkPair represents a bundle of physical links between two PNOs at the
cell level. It gives information about the maximum capacity of these physical links. This is used by
the path searching algorithm .
The interPnoTopologicalSubnetworkPair is bidirectional.
The inherited attributes aEndPoint and zEndPoint contain the two subnetwork object instances
associated to the SubnetworkPair. As it was stressed in subclause 9.5.1  the aEnd and zEnd
correspond to the PNOs taken in the alphabetical order.
The trailList attribute inherited from subNetworkPair will always be empty.
The Notifications from this object shall be broadcasted to every PNOs in the Network.
The name type of the attribute subNetworkPairId, inherited from subnetworkPair, is an pString.
Any change in the attribute listOfAtmAccessPointPairResources shall be reported.
OperationalState Disabled means that a failure has occurred in the complete bundle of Inter-PNO
Physical Links. Enabled means that this bundle is in service."
;;
ATTRIBUTES
"CCITT ITU-T Recommendation X.721 [2] (1992)|ISO/IEC 10165-2: (1992)":operationalState GET,
listOfAtmAccessPointPairResources GET;
;;

REGISTERED AS {xatmObjectClass 3};
```

PNO VP Subnetwork

```
pnoVpSubnetwork MANAGED OBJECT CLASS

DERIVED FROM "DTS 43316":subnetwork;
CHARACTERIZED BY
pnoVpSubnetworkPackage PACKAGE
BEHAVIOUR pnoVpSubnetworkDefinition BEHAVIOUR
DEFINED AS
"A pnoVpSubnetwork Object represents the complete Subnetwork of a certain Operator, from a
topological point of view, at the VP layer.
A PNO VP Subnetwork offers external interfaces to other PNO VP Subnetworks through PNO NW ATM Access
Points.
From a Connectivity point of view, pnoVpSubnetworks are crossed by Subnetwork Connections.
The pnoVpSubnetwork manages the establishment (reservation) and release of Subnetwork Connections.
So pnoVpSubnetworkConnection object instances are created when connections are requested.
The establishment of a subnetwork connection is performed by the action
reservePnoVpSubnetworkConnection. In the case of a positive result the SuccessfulReserve information
has to provide the far-endVPCTPId, the far-endAPId and the far-endAssociatedAPId if it comes from a
Transit Subnetwork. In the case of a successful reservation in a Z Subnetwork the first part of the
CHOICE in SuccessfulReserve may be returned, instead of the zAddress. In this case the far-
endassociatedAPId could be filled with a „NULL" pString.

The Actions inherited from subNetwork:
addToSubNetworkConnection,
deleteFromSubNetworkConnection,
setupSubNetworkConnection,
releaseSubNetworkConnection
are not used.
The Notifications of this object shall be broadcasted to every PNO in the Network.
The Action reservePnoVpSubnetworkConnection is performed by the Initiating PNO with the Transit and
Z PNOs.

The Action giveAvailableLinks can be performed by the initiating PNO with the Transit Pno’s and the
Z PNO.
The Action chaeckUser is performed by the initiating PNO with the Z PNO.
The operationalState has two possible values: Disabled or Enabled. Disabled means that a failure has
occurred in the PNO VP Subnetwork. Enabled means that the Subnetwork is in service.
."
;;
ACTIONS
giveAvailableLinks,
checkUser,
reservePnoVpSubnetworkConnection,
releasePnoVpSubnetworkConnection;
```

```
;;

REGISTERED AS {xatmObjectClass 4};
```

PNO NW ATM Access Point

```
pnoNWAtmAccessPoint MANAGED OBJECT CLASS

DERIVED FROM "CCITT ITU-T Recommendation X.721 [2] (1992) | ISO/IEC 10165-2: 1992:92":top;
CHARACTERIZED BY
"ITU-T Recommendation M.3100 [1]:92":tmnCommunicationAlarmInformationPackage,
"ITU-T Recommendation M.3100 [1]:92":stateChangeNotificationPackage,
"ITU-T Recommendation M.3100 [1]:92":attributeValueChangeNotificationPackage,
pnoNWAtmAccessPointPackage PACKAGE
BEHAVIOUR pnoNWAtmAccessPointDefinition BEHAVIOUR
DEFINED AS
"A pnoNWAtmAccessPoint object represents either an endpoint of a physical link at the cell level
between two PNOs or a User Network Interface (UNI).
For UNI’s the value of attribute associatedSubnetworkPairId is always „UNI".
When a failure is detected on this Access Point or on the associated Physical Link a failure
notification indication is given across the Xcoop.
The Notifications from this object shall be sent to every PNO Subnetwork.
Changes in attribute maxNumVPIBitsSupported are reported by an attributeValueChangeNotification."
;;
ATTRIBUTES
pnoNWAccessPointId GET,
associatedSubNetworkPairId GET,
"CCITT ITU-T Recommendation X.721 [2] (1992)|ISO/IEC 10165-2: (1992)":operationalState GET,
"ITU-T Recommendation I.751:maxNumVPIBitsSupported GET;
;;

REGISTERED AS {xatmObjectClass 5};
```

8.5.5 Continuity Check Fragment

```
pnoBidirectionalContinuityMonitor MANAGED OBJECT CLASS

DERIVED FROM "ITU-T Rec.I.751":bidirectionalContinuityMonitor;
;;

REGISTERED AS { xatmObjectClass 6 };
```

8.6 Package Definitions

8.6.1 Attribute Value Change Notification Package

This package is defined in ITU-T Recommendation M.3100 [1].

8.6.2 Create Delete Notification Package

This package is defined in ITU-T Recommendation M.3100 [1].

8.6.3 State Change Notification Package

This package is defined in ITU-T Recommendation M.3100 [1].

8.6.4 TMN Communication Alarm Information Package

This package is defined in ITU-T Recommendation M.3100 [1].

8.6.5 cancelVpNetworkConnectionNotification Package

```
cancelVpNetworkConnectionNotificationPkg PACKAGE

BEHAVIOUR
cancelVpNetworkConnectionNotificationPkgBehaviour BEHAVIOUR
DEFINED AS
"This package allows the Z PNO to notify the Initiating PNO to release the VP Network Connection."
;;
```

```
NOTIFICATIONS
cancelVpNetworkConnectionNotification;
REGISTERED AS { xatmPackage 1 };
```

8.6.6 vpTestStatePackage

```
vpTestStatePackage PACKAGE

BEHAVIOUR
vpTestStatePkgBehaviour BEHAVIOUR
DEFINED AS
"This package gives an indication if it is allowed or not to perform tests on the PNO VP
Connection."
;;
ATTRIBUTE
vpTestState GET-REPLACE ;

REGISTERED AS { xatmPackage 2 };
```

8.6.7 flowDirectionPackage

This package is defined in ITU-T Recommendation I.751 [5].

8.7 Attribute Definitions

8.7.1 Associated SubNetwork Pair Id

```
associatedSubNetworkPairId ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.NameType;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    associatedSubnetworkPairIdBehaviour BEHAVIOUR
DEFINED AS "This attribute contains a pointer to the SubNetworkPair to whom the PNO NW ATM Access
Point belongs to.";;

REGISTERED AS {xatmAttribute 1};
```

8.7.2 Backward QoS Class

```
backwardQoSClass ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.VpQoSClass;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    backwardQoSClassBehaviour BEHAVIOUR
DEFINED AS "This attribute contains an indication of the QoS class in the backward direction of the
VPC. The class corresponds to specified values of delay, error rate and protection level values";;

REGISTERED AS {xatmAttribute 2};
```

8.7.3 Change Reservation Information

```
changeReservationInformation ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.ChangeReservationInfo;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    changeReservationInformationBehaviour BEHAVIOUR
DEFINED AS "This attribute contains the new parameters for modification of the PNO VP Subnetwork
Connection. These parameters are kept while the modification is not activated by the Initiating PNO
using the activateChange action.
Before the change is actually performed by this action, the new parameters are stored by means of
REPLACE operation on this attribute.
The cancel of the modification is done through the operation REPLACE-BY-DEFAULT on this
attribute.";;

REGISTERED AS {xatmAttribute 3};
```

8.7.4 Forward QoS Class

```
forwardQoSClass ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.VpQoSClass;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    forwardQoSClassBehaviour BEHAVIOUR
DEFINED AS "This attribute contains an indication of the QoS class in the forward direction of the
VPC. The class corresponds to specified values of delay, error rate and protection level values";;

REGISTERED AS {xatmAttribute 4};
```

8.7.5 Initiating Pno Subnetwork Id

```
initiatingPnoSubnetworkId ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.InitiatingPnoSubnetworkId;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    initiatingPnoSubnetworkIdBehaviour BEHAVIOUR
DEFINED AS "This attribute identifies the Initiating PNO Subnetwork, and together with the
initiatingVpConnectionId, identifies the end-to-end VP Connection.
The use of pString for this attribute is preferred"
;;

REGISTERED AS {xatmAttribute 5};
```

8.7.6 Intiating Vp Connection Id

```
initiatingVpConnectionId ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.InitiatingVpConnectionId;
    MATCHES FOR EQUALITY, SUBSTRINGS;
    BEHAVIOUR
    initiatingVpConnectionIdBehaviour BEHAVIOUR
DEFINED AS "This attribute contains the identifier that the Initiating PNO Subnetwork assigns to the
VP Connection, and together with the initiatingPnoSubnetworkId, identifies the end-to-end VP
Connection."
;;

REGISTERED AS {xatmAttribute 6};
```

8.7.7 List of ATM Access Point Pair Resources

```
listOfAtmAccessPointPairResources ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.ListOfAtmAccessPointPairResources;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    listOfAtmAccessPointPairResourcesBehaviour BEHAVIOUR
DEFINED AS "This attribute contains the list of the NW ATM Access Points contained within the Inter
PNO Topological Subnetwork Pair, with their associated Cell Rate and QoS Class.
The aPnoAtmAccessPointId contains the Id of the access point that belongs to the network in
aEndPoint. The same applies to zPnoAtmAccessPointId and zEndPoint.
The maxAtoZ bandwith refers to the direction from the aPnoAtmAccessPoint to the zPnoAtmAccessPoint.
The maxZtoA bandwith refers to the other direction";;

REGISTERED AS {xatmAttribute 7};
```

8.7.8 Max Num VPI Bits Supported

It is defined in the ITU-T Recommendation I.751 [5].

8.7.9 Operational State

It is defined in the ITU-T Recommendation X.721 [2].

8.7.10 PNO NW Access Point Id

```
pnoNWAccessPointId ATTRIBUTE
```

```
    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.NameType;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    pnoNWAccessPointIdBehaviour BEHAVIOUR
DEFINED AS "This attribute is used for naming pnoNWAtmAccessPoint objects.";;

REGISTERED AS {xatmAttribute 8};
```

8.7.11 VPCTP Id

It is defined in the ITU-T Recommendation I.751 [5]

8.7.12 VP Schedulers

```
vpSchedulers ATTRIBUTE

    WITH ATTRIBUTE SYNTAX   ASN1XatmModule.VpSchedulers;
    MATCHES FOR EQUALITY;
    BEHAVIOUR
    vpSchedulersBehaviour BEHAVIOUR
DEFINED AS "This attribute includes a scheduler specifying a number of time intervals on 24 hour
time-of-day clock, pertaining to selected days of the week for each month. The traffic descriptor is
associated with this scheduler allowing the change of the bandwith on a schedule basis..";;

REGISTERED AS {xatmAttribute 9};
```

8.7.13 vpTestState

```
vpTestState ATTRIBUTE

WITH ATTRIBUTE SYNTAX   ASN1XatmModule.VpTestState;
MATCHES FOR EQUALITY;
BEHAVIOUR
vpTestStateBehaviour BEHAVIOUR
DEFINED AS "This attribute gives an indication if  it is allowed or not to perform tests on the PNO
VP Connection."
;;

REGISTERED AS { xatmAttribute 10 };
```

8.7.14 continuityMonitorId

Defined in ITU-T Recommendation I.751 [5].

8.7.15 sinkCCMecanismActive

Defined in ITU-T Recommendation I.751 [5].

8.7.16 sourceCCMechanismActive

Defined in ITU-T Recommendation I.751 [5].

8.8 Attribute Group Definitions

No attribute groups were identified.

8.9 Parameter Definitions

No parameters were identified.

8.10 Notification Definitions

8.10.1 Cancel VP Network Connection Notification

```
cancelVpNetworkConnectionNotification NOTIFICATION

BEHAVIOUR
cancelVpNetworkConnectionNotificationBehaviour BEHAVIOUR
DEFINED AS "This notification is issued by the Z PNO to ask to the initiating PNO to release the VP
Network Connection."
;;
WITH INFORMATION SYNTAX ASN1XcoopModule.CancelVpNetworkConnectionNotification;
REGISTERED AS { xatmNotification 1 };
```

8.11 Action Definitions

8.11.1 Activate Change

```
activateChange ACTION

    BEHAVIOUR
activateChangeBehaviour BEHAVIOUR
DEFINED AS "This action is requested by the Initiating PNO when it wants to activate a change
reservation already confirmed by all the PNO Subnetworks in the path.
The activateChange action consists of changing the parameters of the current activation, in
accordance with the previous changeReservation procedure.
This action entails to replace the vpSchedulers attribute  with the proper component of the
changeReservationInformation attribute. Then, the changeReservationInformation attribute will be
replaced with its default value. All these operations are internal to the concerned PNO, so these
changes of the attribute values should be notified by using the attributeValueChangeNotification."
;;
    MODE CONFIRMED;

    WITH REPLY SYNTAX ASN1XatmModule.ActivateChangeResult;

REGISTERED AS{xatmAction 1} ;
```

8.11.2 Check User

```
checkUser ACTION

    BEHAVIOUR
checkUserBehaviour BEHAVIOUR
DEFINED AS "The objective of this action is to check if the Z User is willing to and is able to
accept the proposed VP Connection.

The meaning of the different check user causes is as follows:

. bandwidthNotAvailable : the destination user is not able to support the band width(s) required in
the schedulers.

. userNotAvailable : it includes the following cases:
- none-existing user,
- user availability status with the value notAvailable,
- user operational state with the value disabled,
- user administrative state with the value locked.
All of this cases are internal to each PNO (not visible from Xcoop).

. userNotCompatible : it includes the following case :
- with regard to a list of Origin users which the destination user does not want to have any
connection. "
;;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX ASN1XatmModule.CheckUserInformation;
    WITH REPLY SYNTAX ASN1XatmModule.CheckUserResult;

REGISTERED AS{xatmAction 2} ;
```

8.11.3 Give Available Links

```
giveAvailableLinks ACTION
```

```
    BEHAVIOUR
giveAvailableLinksBehaviour BEHAVIOUR
DEFINED AS "The result of this Action is a list of  PNOs that are adjacent to the PNO that receives
this request and have sufficient cell rate available on their links with the receiving PNO to
support the proposed VP Connection."
;;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX     ASN1XatmModule.GiveAvailableLinksInformation;
    WITH REPLY SYNTAX ASN1XatmModule.GiveAvailableLinksResult;

REGISTERED AS{xatmAction 3} ;
```

8.11.4 Release PNO VP Subnetwork Connection

```
releasePnoVpSubnetworkConnection ACTION

    BEHAVIOUR
releasePnoVpSubnetworkConnectionBehaviour BEHAVIOUR
DEFINED AS "This action is performed by the Initiating PNO requesting the clearing down of the VP
subnetwork connection. This will delete the pnoVpSubnetworkConnection object instance that makes up
the connection. The related pnoVPCTP object instance(s) that are not in use by other VP subnetwork
connections are deleted too."
;;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX     ASN1XatmModule. ReleaseSubNetworkConnectionInformation;
    WITH REPLY SYNTAX ASN1XatmModule. ReleaseSubNetworkConnectionResult;

REGISTERED AS{xatmAction 4} ;
```

8.11.5 Reserve PNO VP Subnetwork Connection

```
reservePnoVpSubnetworkConnection ACTION

    BEHAVIOUR
reservePnoVpSubnetworkConnectionBehaviour BEHAVIOUR
DEFINED AS "This action is performed by the Initiating PNO requesting a connection reservation to
this PNO Subnetwork, between two of its endpoints (pnoNWAtmAccessPoints). The result of this action
is the acceptance or reject of the connection reservation request (regarding the start time, the
stop time and eventually the periodicity requested).
If the connection reservation is rejected, the reason is returned. The meaning of the different
causes for rejection is as follows:
```

nearEndQosNotAvailable(1) : With the *nearEndPnoSubnetworkId* the Agent will find the corresponding *interPnoTopologicalSubnetworkPair* Object Instance. Within this instance, the Agent will look for the element containing the nearEndApId in its *listOfAtmAccessPointPairResources* attribute.

Once the Agent has found the proper element in the list, it must check whether the forwardQoSClass and BackwardQoSClass fields in the *reserveInformation* can be provided by the *nearEndAPId* according to the *atmPathQoS* field in the *listOfAtmAccessPointPairResources* attribute.

scheduleNotAvailable(2) : the agent can NOT provide the *vpConnection* with the specified *vpScheduler*.

As described in the previous cause, once the Agent has found the proper element in the *listOfAtmAccessPointPairResources* attribute, it must check the following :

At the nearEnd side (CASE 1) :

- Whether the requested AtoZTrafficDescriptor of each slot of the VpScheduler fits in the current available incoming bandwidth. Note that the current available incoming bandwidth is internal to the agent (not visible from the Xcoop interface) and must be obtained by means of the max. incoming bandwidth and the previous reservations performed on this nearEndAPId.

- Whether the requested ZtoATrafficDescriptor of each slot of the VpScheduler fits in the current available outgoing bandwidth. Note that the current available outgoing bandwidth is internal to the agent (not visible from the Xcoop interface) and must be obtained by means of the max. outgoing bandwidth and the previous reservations performed on this nearEndAPId.

If the nearEndAPId does not fulfil the two above conditions, the Reservation Action Response will contain an unsuccessful response with the scheduleNotAvailable reserve cause.

At the far End side (case 2: TRANSIT), the conditions to be checked are :

- The requested AtoZTrafficDescriptor of each slot of the VpScheduler fits in the current available outgoing bandwidth on the FarEndAPId which is being checked. Note that the current available outgoing bandwidth is internal to the agent (not visible from the Xcoop interface) and must be obtained by means of the max. outgoing bandwidth and the previous reservations performed on this farEndAPId which is being checked.

- The requested ZtoATrafficDescriptor of each slot of the VpScheduler fits in the current available incoming bandwidth on the FarEndAPId which is being checked. Note that the current available incoming bandwidth is internal to the agent (not visible from the Xcoop interface) and must be obtained by means of the max. incoming bandwidth and the previous reservations performed on this farrEndAPId which is being checked.

If there is no element (link) in the ListOfAtmAccessPointPairResources attribute to fulfil the two above conditions, the Reservation Action Response will contain an unsuccessful response with the ScheduleNotAvailable reserve cause.

At the far End side (case 3: DESTINATION*)*, the conditions to be checked are :

- The requested AtoZTrafficDescriptor of each slot of the VpScheduler fits in the current available outgoing bandwidth on the UNI resource. Note that the current available outgoing bandwidth on this resource is internal to the agent (not visible from the Xcoop interface) and can be obtained by means of the max. outgoing bandwidth on this UNI resource on the Q3 level and the previous reservations performed on this UNI associated to the destinationAddress.

- The requested ZtoATrafficDescriptor of each slot of the VpScheduler fits in the current available incoming bandwidth on the UNI. Note that the current available incoming bandwidth is internal to the agent (not visible from the Xcoop interface) and can be obtained by means of the max. incoming bandwidth on this UNI resource on the Q3 level and the previous reservations performed on this UNI associated to the Destination Address.

If the UNI associated to the Destination Address does not fulfil the two above conditions, the Reservation Action Response will contain a negative response with the ScheduleNotAvailable reserve cause.

. nearEndVpiBusy(3) : the *nearEndVPCTPId* (see ReserveInformation ASN.1 definition) is already used by an another vpConnection during the specified slot times.

. zVpiBusy(4) : the *zVPi* (see ReserveInformation ASN.1 definition) is already used during the specified slot times.

. nearEndVpiOutOfRange(5) : This happens when the maximum range of subnetwork connections on the pnoNWAccessPoint Object Instance (designed by *nearEndAPId*) have previously been allocated.

. zVpiOutOfRange(6) : This happens when the maximum range of subnetwork connections on the pnoNWAccessPoint Object Instance at USER side have previously been allocated. .

. nearEndSNUnknown(7) : the *nearEndPnoSubnetworkId* (see ReserveInformation ASN.1 definition) is unknown.

. farEndSNUnknown(8) : the *farEndPnoSubnetworkId* (see ReserveInformation ASN.1 definition) is unknown.

. userNotAvailable(9) : it includes the following cases :

- none-existing user,
- user availability status with the value *notAvailable*,
- user operational state with the value *disabled*,
- user administrative state with the value *locked*.

. userNotCompatible(10) : it includes the following case :

- with regard to a list of Origin users which the destination user does not want to have any connection.

. nearEndAPisUnknown(11) : the *nearEndAPId* (see ReserveInformation ASN.1 definition) is unknown.

. modeNotAvailable(12) : the *mode* (see ReserveInformation ASN.1 definition) is not *pointToPoint(0).*

. initiatingPnoSNUnknown(13) : the PNO indicated by *initiatingPnoSubnetworkId* (see ReserveInformation ASN.1 definition) is unknown.

. farEndQosNotAvailable(14) :

At the far End side (case 1: TRANSIT, the conditions to be checked are :

- Whether the forwardQoSClass and BackwardQoSClass fields in the ReserveInformation can be provided by the pnoNWAtmAccess Point Object Instance identified in the element being checked according to the atmPathQoS field in the ListOfAtmAccessPointPairResources attribute.

If there are no elements in the ListOfAtmAccessPointPairResources attribute that fulfil the above condition, then the Agent will response with an unsuccessful response with the farEndQoSNotAvailable reserve cause.

At the far End side (case 2: DESTINATION Z net), the conditions to be checked are :

- Whether the forwardQoSClass and BackwardQoSClass fields in the ReserveInformation can be provided by the UNI resource according to its Quality of Service at the Q3 network level.

. refused(15) : An agent PNO can refuse the reservation if the initiating PNO is not the A PNO

In case of a ***Z*** PNO it is possible to define the zVPi within the reserveInformation. If this requested zVPi can not be provided, the reserveCause vpiNotAvailable is also to be responded.
If the connection reservation is accepted, a pnoVpSubnetworkConnection object is created."

```
;;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX ASN1XatmModule.ReserveInformation;
    WITH REPLY SYNTAX   ASN1XatmModule.ReserveResult;

REGISTERED AS {xatmAction 5};
```

8.11.6 controlCC

Defined in ITU-T Recommendation I.751 [5]

8.12 Name Binding Definitions

8.12.1 interPnoTopologicalSubnetworkPair-pnoVpSubnetwork

```
interPnoTopologicalSubnetworkPair-pnoVpSubnetwork NAME BINDING

    SUBORDINATE OBJECT CLASS interPnoTopologicalSubnetworkPair;
    NAMED BY
    SUPERIOR OBJECT CLASS pnoVpSubnetwork;
    WITH ATTRIBUTE "I-ETS 300 653 [3] ": subnetworkPairlId;
BEHAVIOUR
interPnoTopologicalSubnetworkPair-pnoVpSubnetworkBehaviour BEHAVIOUR
DEFINED AS
"The interPnoTopologicalSubnetworkPair object is not created or deleted by system management
protocol. An instance of the object is created when  a bundle of physical links between one PNO and
another, comes into existence.
The object is deleted when the underlying resource ceases to exist."
;;

REGISTERED AS {xatmNameBinding 1};
```

8.12.2 pnoNWAtmAccessPoint-pnoVpSubnetwork

```
pnoNWAtmAccessPoint-pnoVpSubnetwork NAME BINDING

    SUBORDINATE OBJECT CLASS pnoNWAtmAccessPoint;
    NAMED BY
    SUPERIOR OBJECT CLASS pnoVpSubnetwork;
    WITH ATTRIBUTE  pnoNWAccessPointId;
BEHAVIOUR
pnoNWAtmAccessPoint-pnoVpSubnetworkBehaviour BEHAVIOUR
DEFINED AS
"The pnoNWAtmAccessPoint object is not created or deleted by system management protocol. An instance
of the object is created when a single physical link, the end point of which this object represents,
comes into existence. This may be on the creation, or at any time during the existence of the
interPnoTopologicalSubnetworkPair object instance associated with a particular pnoNWAtmAccessPoint
object instance.
A pnoNWAtmAccessPoint object instance may be created representing the external connection to an End
User in a Z PNO, in which case there will be no associated subnetworkPairId, as such an object would
not be related to an inter pno physical link.
The object is deleted when the related physical link ceases to exist, or, in the case of a Z PNO,
when the related connection to a User ceases to exist.
The creation and deletion of pnoNWAtmAccessPoint object instances are only notified to other PNOs,
if these instances are associated with interPnoTopologicalSubnetworkPair instance, via the
attributeValueChangeNotification, reporting on changes to the listOfAtmAccessPointPairResources
attribute.
No reports for the creation and deletion of pnoNWAtmAccessPoint object instances serving connections
to users in Z PNOs will be made."
;;

REGISTERED AS {xatmNameBinding 2};
```

8.12.3 pnoVPCTP-pnoNWAtmAccessPoint

```
pnoVPCTP-pnoNWAtmAccessPoint NAME BINDING

    SUBORDINATE OBJECT CLASS pnoVPCTP;
    NAMED BY
    SUPERIOR OBJECT CLASS pnoNWAtmAccessPoint;
    WITH ATTRIBUTE " Rec.I.751 [5] ": vpCTPId;
BEHAVIOUR
pnoVPCTP-pnoNWAtmAccessPointBehaviour BEHAVIOUR
DEFINED AS
"A pnoVPCTP object instance is created by the reservePnoVpSubnetworkConnection action, when a VPI is
allocated on a physical link between two PNOs. The object instance is deleted by the action
releasePnoVpSubnetworkConnection, when it is not used by other pnoVpSubnetworkConnections."
;;
REGISTERED AS {xatmNameBinding 3};
```

8.12.4 pnoVpSubnetworkConnection-pnoVpSubnetwork

```
pnoVpSubnetworkConnection-pnoVpSubnetwork NAME BINDING

    SUBORDINATE OBJECT CLASS pnoVpSubnetworkConnection;
    NAMED BY
    SUPERIOR OBJECT CLASS pnoVpSubnetwork;
    WITH ATTRIBUTE subnetworkConnectionId;
BEHAVIOUR
pnoVpSubnetworkConnection-pnoVpSubnetworkBehaviour BEHAVIOUR
DEFINED AS
"A pnoVpSubnetworkConnection object instance is created by the reservePnoVpSubnetworkConnection
action, when a VPI is allocated on a physical link between two PNOs. The object instance is deleted
by the action releasePnoVpSubnetworkConnection."
;;

REGISTERED AS {xatmNameBinding 4};
```

8.12.5 pnoBidirectionalContinuityMonitor-pnoVPCTP

```
pnoBidirectionalContinuityMonitor-pnoVPCTP NAME BINDING

SUBORDINATE OBJECT CLASS pnoBidirectionalContinuityMonitor;
NAMED BY SUPERIOR OBJECT CLASS pnoVPCTP;
WITH ATTRIBUTE   continuityMonitorId;
CREATE
WITH-REFERENCE-OBJECT, WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE
DELETES-CONTAINED-OBJECTS;
```

```
REGISTERED AS { xatmNameBinding 5 };
```

8.13 ASN.1 Module

```
ASN1XatmModule {ccitt(0) identified-organization (4) etsi (0) EN300820-1 (820) informationModel (0)
asn1Module (2) asn1TypesModule (0)}

DEFINITIONS IMPLICIT TAGS::=

BEGIN

IMPORTS

    NameType
    FROM ASN1DefinedTypesModule {ccitt (0) recommendation (0) m (13) gnm (3100)
    informationModel (0) asn1Modules (2) asn1DefinedTypeModule (0)}


    VpiValue
    PeakCellRate,
    SustainableCellRate,
    CDVTolerance,
    MaxBurstSize
    FROM AtmMIBMod {itu-t(0) recommendation(0) i(9) atmm(751) informationModel(0) asn1Module(2)
atm(0)}


    ReleaseSubNetworkConnectionResult,
    Mode

    FROM I-ETS300 653 [3] {ccitt (0) identified-organization (4) etsi (0) ets (653)
    informationModel (0) asn1Module (2) i-ets300 653 [3] (0)}


    StopTime, Time24
    FROM Attribute-ASN1Module {joint-iso-ccitt ms (9) smi (3) part2 (2) asn1Module (2) 1}


xatmInfoModel OBJECT IDENTIFIER::= {ccitt(0) identified-organization(4) etsi(0) EN300820-1 (820)
informationModel(0)}

xatmSpecificExtension OBJECT IDENTIFIER::= {xatmInfoModel standardSpecificExtension(0)}

xatmObjectClass OBJECT IDENTIFIER::= {xatmInfoModel managedObjectClass(3)}

xatmPackage OBJECT IDENTIFIER::= {xatmInfoModel package(4)}

xatmNameBinding OBJECT IDENTIFIER::= {xatmInfoModel nameBinding(6)}

xatmAttribute OBJECT IDENTIFIER::= {xatmInfoModel attribute(7)}

xatmAction OBJECT IDENTIFIER::= {xatmInfoModel action(9)}

xatmNotification OBJECT IDENTIFIER::= {xatmInfoModel notification(10)}


ActivateChangeResult::= Result

AtmPathQoS::= INTEGER (0..99)


BidirectionalTrafficDescriptor::= SEQUENCE {
        aToZ                        TrafficDescriptor,
        zToA                        TrafficDescriptor}


CancelVpSubNetworkConnectionNotification::= SEQUENCE {
        initiatingPnoSubnetworkId   InitiatingPnoSubnetworkId,
        initiatingVpConnectionId    InitiatingVpConnectionId,
             disconnectCause             DisconnectCause }


ChangeReservationInfo::= VpSchedulers

CheckUserInformation::= SEQUENCE {
        initiatingPnoSubnetworkId           InitiatingPnoSubnetworkId,
        initiatingVpConnectionId            InitiatingVpConnectionId,
        vpSchedulers                            VpSchedulers,
        aAddress                                E164Address,        -- This corresponds to the
address of
```

```
                                                                    -- the A side
        zAddress                                E164Address         -- This corresponds to the
address of
                                                                    -- the Z side
        }


CheckUserResult::=  CHOICE {
        checkingOk              NULL,
        checkingNotok           UserCause }



DailySchedule::= SEQUENCE OF DaySlot

DaySlot::= SEQUENCE {
        slotBegin Time24 ,
        slotEnd   Time24 ,
        bandwidth BidirectionalTrafficDescriptor}


defaultChangeReservationInfo        ChangeReservationInfo::= {
        startTime                       NULL,
        stopTime                        NULL,
        aToZTrafficDescriptor           NULL,
        zToATrafficDescriptor           NULL

        }


Digit::= INTEGER (0..9)

DisconnectCause ::= ENUMERATED  {normal(0), unrecoverableFailure(1)}

E164Address::= SEQUENCE SIZE (1 .. 15) OF Digit


GiveAvailableLinksInformation::= SEQUENCE {
        initiatingPnoSubnetworkId           InitiatingPnoSubnetworkId,
        initiatingVpConnectionId            InitiatingVpConnectionId,
        aPnoId                          [0] NameType  OPTIONAL,
        vpSchedulers                        VpSchedulers,
        aToZQoSClass                        VpQoSClass, -- A to Z means from the nearend to the far-
end
        zToAQoSClass                        VpQoSClass  -- Z to A means from the far-end to the
nearend}


GiveAvailableLinksResult ::= CHOICE {
        null                    NULL,
        listOfSubnetworks       SET OF NameType}


InitiatingPnoSubnetworkId       NameType

InitiatingVpConnectionId        NameType


ListOfAtmAccessPointPairResources::= SET OF SEQUENCE {
                aPnoAtmAccessPointId        NameType,
                zPnoAtmAccessPointId        NameType,
                maxAtoZBandwith             INTEGER,
                maxZtoABandwith             INTEGER,
                atmPathQoS                  AtmPathQoS}


MonthDay::= INTEGER (1..31)

MonthlySchedule::= SEQUENCE OF MonthSlot

MonthSlot::= SEQUENCE {
        slotBegin TimeMonth ,
        slotEnd   TimeMonth ,
            bandwidth BidirectionalTrafficDescriptor}


OccasionalSchedule::= SEQUENCE OF OccasionalSlot

OccasionalSlot::= SEQUENCE {
        slotBegin StartTime ,
        slotEnd   StopTime ,
        bandwidth BidirectionalTrafficDescriptor}
```

```
ReleaseSubNetworkConnectionInformation::= CHOICE {
        snc     ObjectInstance,
        SEQUENCE {
            initiatingPnoSubnetworkId   InitiatingPnoSubnetworkId,
             initiatingVpConnectionId   InitiatingVpConnectionId } }


ReserveCause::= ENUMERATED, nearEndQosNotAvailable(1),
                        scheduleNotAvailable(2), nearEndVpiBusy(3),  zVpiBusy(4),
                        nearEndVpiOutOfRange(5),  zVpiOutOfRange(6),
                        nearEndSNUnknown(7),  farEndSNUnknown(8),
                        userNotAvailable(9), userNotCompatible(10),
                        nearEndAPisUnknown (11), modeNotAvailable (12),
                 initiatingPnoSNUnknown (13),  farEndQosNotAvailable(14),
                 refused (15) }



ReserveInformation::= SEQUENCE {
        initiatingPnoSubnetworkId           InitiatingPnoSubnetworkId,
        initiatingVpConnectionId            InitiatingVpConnectionId,
        aPnoId                          [1] NameType  OPTIONAL,
        CHOICE {
        aAddress                    [0]     E164Address,
        SEQUENCE {
            nearEndAPId                         NameType,
            nearEndVPCTPId                      VpiValue,
            nearEndPnoSubnetworkId              NameType}},
        CHOICE {
        farendPnoSubnetworkId           NameType,
        SEQUENCE {  aAddress            E164Address,
                    zAddress            E164Address}},
        zVPi    [0]                     VpiValue OPTIONAL,
        aToZQoSClass                    VpQoSClass,
        zToAQoSClass                    VpQoSClass,
        configurationType               Mode,
        vpSchedulers                    VpSchedulers}


ReserveResult::= CHOICE {
        unsuccessfulResult          [0] ReserveCause,
        successfulResult            [1] SuccessfulReserve}


Result::= ENUMERATED {successful(0), unsuccessful(1)}

StartTime::= StopTime


SubnetworkConnectionId::=   NameType


SuccessfulReserve::= CHOICE {
        SEQUENCE {
            far-endVPCTPId                  VpiValue ,
            far-endAPId                     NameType,
            far-endassociatedAPId           NameType,}
            zAddress                    [0] E164Address}

TimeMonth ::= SEQUENCE {
        monthDay MonthDay ,
            time Time24}


TimeWeek ::= SEQUENCE {
        weekDay WeekDay ,
        time Time24}


-- This Typedefinition of TrafficDescriptor does not align with
-- „Descriptor" as defined in I.751.
-- In future versions ot the X-interface standard ETSI shall
-- consider migration to the Type that is used in I.751.

TrafficDescriptor::= CHOICE { NULL,
SEQUENCE {
    peakCellRate            PeakCellRate,
    cDVTolerancePCR         CDVTolerance,
    sustainableCellRate [0] SustainableCellRate OPTIONAL,
    cDVToleranceSCR     [1] CDVTolerance        OPTIONAL
    maxBurstSize            MaxBurstSize        OPTIONAL}}
```

```
UserCause   ::= ENUMERATED {bandwithNotAvailable(0), userNotAvailable(1),
                    userNotCompatible(2)}

VpQoSClass  ::= INTEGER (0..99)


VpTestState::= ENUMERATED{inhibitTest(0), allowTest(1) }
VpSchedulers    ::= SEQUENCE {
        startTime           StopTime
        stopTime            StopTime
        CHOICE {
                    durationScheduling          [0] BidirectionalTrafficDescriptor,
                    dailyScheduling             [1] DailySchedule,
                    weeklyScheduling            [2] WeeklySchedule
                    monthlyScheduling           [3] MonthlySchedule
                    occasionalScheduling        [4] OccasionalSchedule}}


WeekDay ::= ENUMERATED {
        sunday (0) ,
        monday (1) ,
        tuesday (2) ,
        wednesday (3) ,
        thursday (4) ,
        friday (5) ,
        saturday (6)}


WeeklySchedule ::= SEQUENCE OF WeekSlot

WeekSlot ::= SEQUENCE {
        slotBegin TimeWeek ,
        slotEnd   TimeWeek ,
            bandwidth BidirectionalTrafficDescriptor}

END
```

Annex A (informative): About the influence of timers on the X-interface communication

Over the X-interface, the responsetime to a reserve-request can be very long, due to the fact that some agents might need a long time to carry out the CMISE-indication. It is obvious that a manager cannot wait forever for the result of his request.

This annex describes how a manager can cope with this by using a timer.

As neither Common Management Information Protocol (CMIP) nor the rest of the OSI-stack have timers defined that guard the maximum responsetime, the application (the Manager or the Agent) will have to guard it.

Most applications will automatically end an association after a relatively short time.

To guard the long response-times, the Manager can apply an additional internal timer (internal = "not visible on the X-interface").

So, the next cases can be considered (a, b, c, d):

a) The response-time is relatively short ("real time"), no communications-error:

   this is the normal case that is covered by the present model.

b) The response-time is usually relatively short, but a communication-error causes the association (connection) to end, during the time that the manager is waiting for the result of his request.

   The Agent cannot use its Common Management Information Service Element (CMISE) response to the original CMISE indication because, due to the lost association, the invoked of the response is now meaningless. The Manager should wait (for a relatively short time) and then there will be two possibilities:

   1) The reservePnoVpSubnetworkConnection has a successfulResult: The Agent uses the object-creation Notification. The Agent is the invoker of a new association that is used for transmitting the Notification; or

   2) The reservePnoVpSubnetworkConnection has an unsuccessfulResult: There is no Notification for the agent to send. The manager, however, is aware of the situation (the association has ended), so he can send the same reserveRequest again in order to get the unsuccessfulResult as a response.

      (Or, first he can do a GET on pnoVpSubnetworkConnection. If he does not see the Instance he can send the same reserveRequest again).

Case b) is covered by the present model too.

c) The response-time is long, no communication-error. This also addresses General Requirements G.7 and G.5:

   1) The internal "Manager-timer" guards the response-time. If the time is passed, the manager can act as if no request has been made at all and bring back his system to the "pre-request" state. To notify the Agent that he gave up waiting the Manager has to send a "releasePnoVpSubnetworkConnection" Action.

d) The response-time is long and a communications-error causes the association to end (or the manager ends it to reduce costs). There are two possibilities:

   1) The reservePnoVpSubnetworkConnection has a successfulResult within the waiting-time of the manager: The agent uses the object-creation Notification. The Agent is the invoker of a new association that is used for transmitting the Notification; or

   2) The reservePnoVpSubnetworkConnection has an unsuccessfulResult: There is no Notification for the agent to send. The manager waits until the time has passed (remember that the waiting has to end some time), he brings back his system in the "pre-request" state and, to be sure, sends a "releasePnoVpSubnetworkConnection" Action to the Agent.

Annex B (informative): Security aspects

With regard to Access Control as seen from a particular Agent's view the next table should be considered:

Table B.1

Objectclass	Access to the instances in accordance with the GDMO definition of the OC for:	Notifications are sent to:
pnoVpSubnetwork	All other PNOs.	All other PNOs
pnoVpSubnetworkConnection (note 1)	Only the PNO that requested the creation of the instances involved. (Initiating PNO)	Only the PNO that requested the creation of the instances involved. (Initiating PNO)
pnoNWAtmAccessPoint (note 2)	All other PNOs.	All other PNOs
interPnoTopologicalSubnetworkPair	All other PNOs.	All other PNOs
pnoVPCTP	Only those PNOs that requested a pnoVpSubnetworkConnection that points to the pnoVPCTP.	Only those PNOs that requested a pnoVpSubnetworkConnection that points to the pnoVPCTP.
NOTE 1: A Manager, trying to read all Instances of pnoVpSubnetworkConection shall only get the Instances in which he is the initiating PNO. Whether his attempt to read all instances should be registered is a matter for "Security Management". NOTE 2: Z nets: After a successful reserve-request has been made in his subnet, an Agent-PNO is free to decide if he wants to create an instance of pnoNWAtmAccessPoint for the user-side of the connection.		

In general, the usage of a "closed user group" is suggested; only PNOs that are in the closed user group are allowed to set up an association with the appropriate PNO.

How this Access Control is implemented is not within the scope of the present document.

Annex C (informative): Mapping between Management Functions, CMISE Services and ObjectClasses

FUNCTIONS	CMISE SERVICES	OBJECT CLASSES / Operations
Activate Change	M-ACTION	pnoVpSubnetworkConnection / activateChange Action
Activate VP Subnetwork Connection	M-SET	pnoVpSubnetworkConnection / administrativeState Attribute
VP Subnetwork Connection Activation Notification (the administrativeState is automatically changed by the agent PNO)	M-EVENT-REPORT	pnoVpSubnetworkConnection / stateChangeNotification
Cancel Change	M-SET	pnoVpSubnetworkConnection / changeReservationInformation Attribute
Cancel VP / VC Subnetwork Connection	M-EVENT-REPORT	pnoVpSubNetworkConnection / cancelVpNetworkConnection Notification
Release VP Subnetwork Connection	M-ACTION	pnoVpSubnetwork / releasePnoVpSubnetworkCon- -nection Action
Change Reservation	M-SET	pnoVpSubnetworkConnection / changeReservationInformation Attribute
Check Available Cell Rate	M-ACTION	pnoVpSubnetwork / giveAvailableLinks Action
Deactivate VP Subnetwork Connection	M-SET	pnoVpSubnetworkConnection / administrativeState Attribute
Destination User Checking	M-ACTION	pnoVpSubnetwork / checkUser Action
Reroute PNO Subnetwork Connection		Not used
Reserve VP Subnetwork Connection	M-ACTION	pnoVpSubnetwork / reservePnoVpSubnetworkCon- -nection Action
Topology Info Changes	M-EVENT-REPORT	interPnoTopologicalSubnetwork- -Pair / createDeleteNotification / attributeValueChangeNotification
Activate Sink for Continuity Check	M-ACTION	pnoBidirectionalContinuityMonitor/ controlContinuity Check Action
Allow F4 flow	M-SET	pnoVpSubnetworkConnection / vpTestState Attribute
Create Bidirectional Continuity Monitor for Continuity Check	M-CREATE	pnoBidirectionalContinuityMonitor / Create
Deactivate Sink for Continuity Check	M-ACTION	pnoBidirectionalContinuityMonitor / controlContinuity Check Action
Delete Bidirectional Continuity Monitor	M-DELETE	pnoBidirectionalContinuityMonitor / Delete
Notification of Bidirectional Continuity Monitor Operational State change	M-EVENT-REPORT	pnoBidirectionalContinuityMonitor / stateChangeNotification Notification
Report of Continuity Check	M-EVENT-REPORT	pnoBidirectionalContinuityMonitor / tmnCommunicationAlarmInformation Notification
Stop F4 flow	M-SET	pnoVpSubnetworkConnection / vpTestState Attribute

Annex D (informative): Future organizational models

This clause describes different organizational models, which might be used in future for establishing X-interface relationships between PNOs. These models should be flexible enough to accommodate many different combinations of interconnected PNOs. However, the choice of which model to use will be determined by agreements between the participating PNOs. The implications that the future organizational models will have on the X-interface model need to be further studied.

Since there are two extreme organizational models and a mixture of those extremes, there can be (at least) three different organizations for a particular VP / VC connection (if there are more than three PNOs involved). These three organizations are shown below:

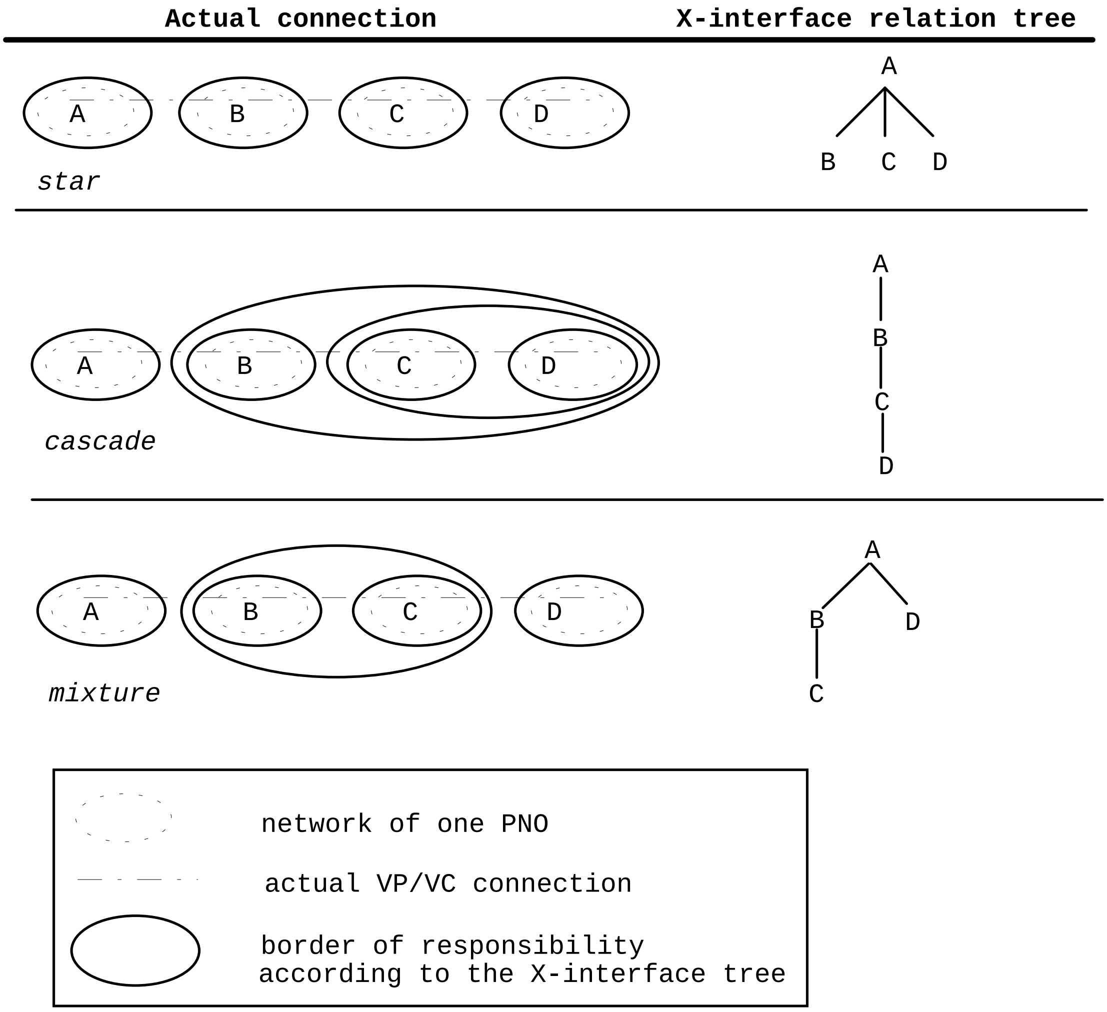


Figure D.1: Organizational models

In the *Star* organization as exemplified in the figure above, PNO A uses the X-interface relation with all the PNOs involved. This means that PNO A has full responsibility for the entire connection and all the other PNOs are only responsible for their own network towards PNO A.

In the *Cascade* organization in this example, PNO A delegates the responsibility for the VP / VC connection outside its network to B; on its turn, B delegates the responsibility for the VP / VC connection outside its network to C, etc. This means that PNO A has responsibility for the entire connection; B, however, is towards PNO A not only responsible for its own network but also for this particular VP / VC connection throughout the networks of PNO C and D; on its turn, PNO C is towards PNO B not only responsible for its own network but also for this particular VP / VC connection throughout the networks of PNO D, etc.

In the "*Mixture*" organization in this particular example (other examples are also possible), PNO A uses an X-interface relationship with both PNO B and PNO D (i.e. a star organization). However, PNO B involves PNO C to reach PNO D. Therefore PNO B is responsible towards PNO A for both its network and the connection throughout the network of PNO C (i.e. cascade organization).

The Consumer/Provider roles, introduced in subclause 4.2, are reflected by the X-interface tree: in the Star example above, PNO has a Consumer role, whereas PNO B, C and D have a Provider role. In the Cascade example above, PNO A has a Consumer role, PNO B has a Provider role towards PNO A, and a Consumer role towards PNO C, etc. In the Mixture example PNO has a Consumer role, whereas PNO B and D have a Provider role towards PNO A; PNO B also has a Consumer role towards PNO C; PNO C has Provider role towards PNO B. Concluding: if a PNOx is directly above a PNOy in the X-interface tree, then PNOx has a Consumer role, and PNOy has a Provider role.

D.1 Responsibility of PNOs regarding a VP / VC connection

This clause denotes which responsibility is required from each PNO involved a particular VP / VC connection. The following rules apply:

Rule 1 A PNO is responsible for the management of a particular VP / VCC within its own PNO Subnetwork and the part of this VP / VCC throughout all the PNO Subnetworks below it in the X-interface tree for this connection (i.e. it is responsible for its subnetwork view). It is responsible for this part of the connection towards its parent in the X-interface tree (if not being the Initiating PNO, who is responsible towards the connection customer).

Rule 2 Besides the responsibility in rule 1, a PNO is also responsible for the ATM Connection over the physical connection from its PNO Subnetwork to the next PNO Subnetwork on the route of a VP / VCC (seen from the A towards the Z).

Rule 3 A PNO can delegate part of its management tasks outside its own PNO subnetwork to one or more other PNOs (these PNOs have a Provider role). This delegation consists of maintaining a requested Quality of Service for this particular connection and reporting about it (via the X-interface to its parent in the X-interface tree).

D.1.1 Examples of application of the responsibility rules

The responsibility rules described in clause D.1 are illustrated by the following scenario description. The scenario starts with the existence of a particular VP / VCC between PNOs A and C via B and a particular X-interface relation tree. The scenario describes the occurrence of 4 failures that need management action; each of them is described in one step of the scenario description. The 4 steps are an example of part of the life cycle of this particular VP / VCC and take place in sequence. The scenario description is illustrated in figures E.2 through E.5.

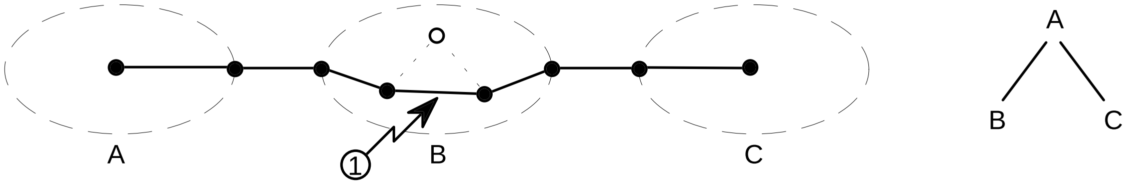


Figure D.2: Step 1 of example scenario

Step 1 A fault has occurred in the VP / VCSC within the PNO subnetwork of PNO B. PNO B reports its Consumer PNO (PNO A) about the failure and indicates that the failure is recoverable. PNO B starts the reconfiguration process and notifies its superior about this (for suppressing alarms). After reconfiguration has been successfully completed, PNO B reports this to its Consumer.

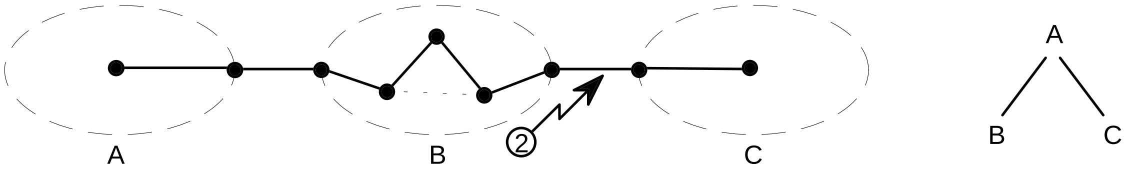


Figure D.3: Step 2 of example scenario

Step 2 A fault has occurred in the inter-PNO VP / VCLC between PNO B and PNO C. Since this is the outgoing inter-PNO VP / VCLC of the PNO subnetwork of PNO B, PNO B is responsible for reporting this failure to its Consumer PNO (PNO A). PNO B reports the failure to PNO A and indicates that it can recover the failure. PNO B performs protection switching on the faulty inter-PNO VP / VCLC and notifies its Consumer about this (for suppressing alarms) and also informs him about the Id of the new physical link in the connection.

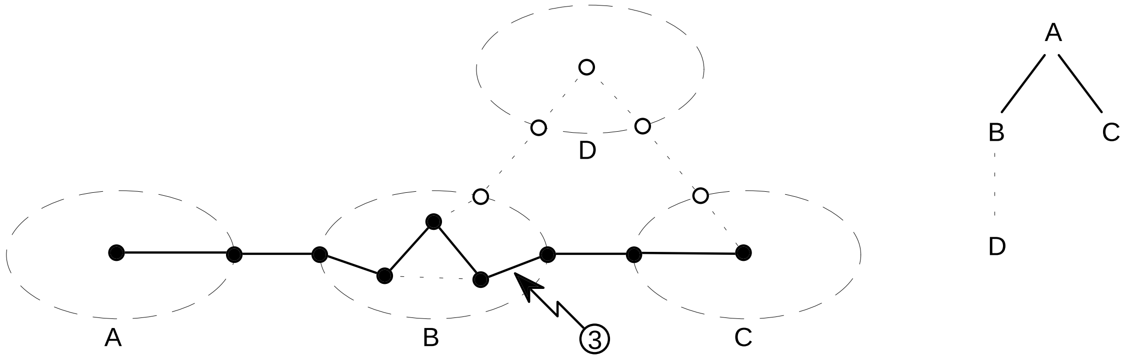


Figure D.4: Step 3 of example scenario

Step 3 A fault has occurred in the VP / VCSC within the PNO subnetwork of PNO B. PNO B reports this to PNO A. Let us assume that protection switching and reconfiguration of the VP / VCSC within the PNO subnetwork of PNO B are not possible. PNO B, however, believes that it can reroute the VP / VCSC via another PNO and indicates this to PNO A. PNO B starts a reconfiguration process, in which it reroutes the VP / VCSC via PNO D, and notifies its Consumer about this (for suppressing alarms). PNO B reserves a VP / VCSC within the PNO subnetwork of PNO D. After successful reservation, PNO D reports back to PNO B (its Consumer) the Id of the physical link between PNO D and PNO C. After reconfiguration has been completed, PNO B reports successful reconfiguration to its Consumer (PNO A) and also informs him about the Id of the new physical link between PNO D and PNO C (the physical link for which PNO B is held responsible). PNO A asks PNO C to cancel the VP / VCSC within the PNO subnetwork of PNO C without releasing the Z user and asks PNO C to reserve a new PNO VP / VCSC according to the new situation. Due to step three in the scenario the X-interface relation turns from *star* to a mixture of *star* and *cascaded*.

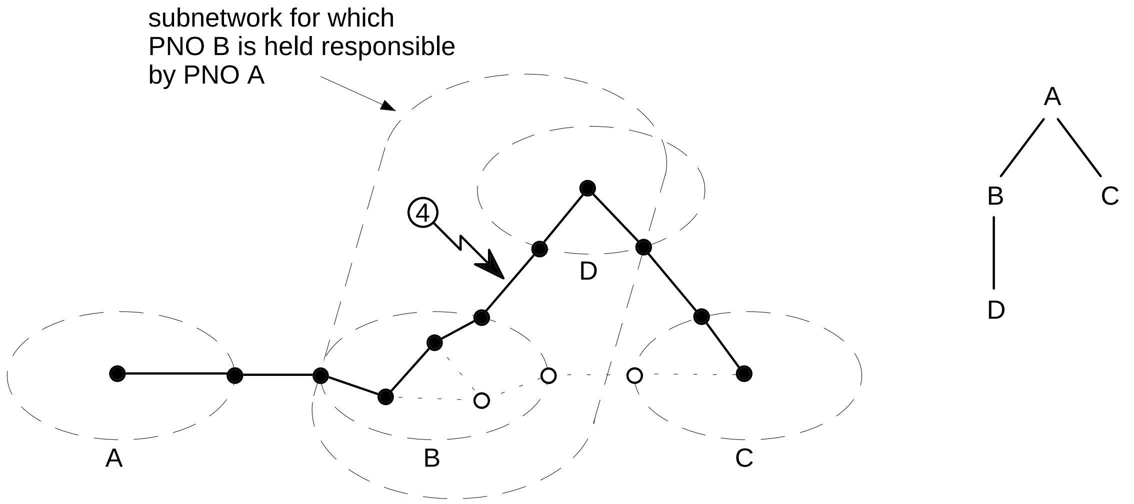


Figure D.5: Step 4 of example scenario

Step 4 A fault has occurred in the physical link between PNO B and PNO D. Because this physical link is now part of the subnetwork of PNO B, PNO B reports the failure to its Consumer (PNO A) as if it were a VP / VCSC failure (instead of a link failure). PNO B indicates that the failure is recoverable. PNO B performs protection switching on the inter-PNO VP / VCLC between PNO B and PNO D and notifies its Consumer about this (for suppressing alarms).

D.2 Scenarios

This clause contains scenarios which apply if the cascaded or mixed mode will be used.

For all scenarios, the following VP / VC connections are possible between PNO A (Initiating) and PNO E (Destination): A-B-C-D-E, A-B-X-Y-D-E and A-B-X-Y-Z-E.

We will distinguish the following cases:

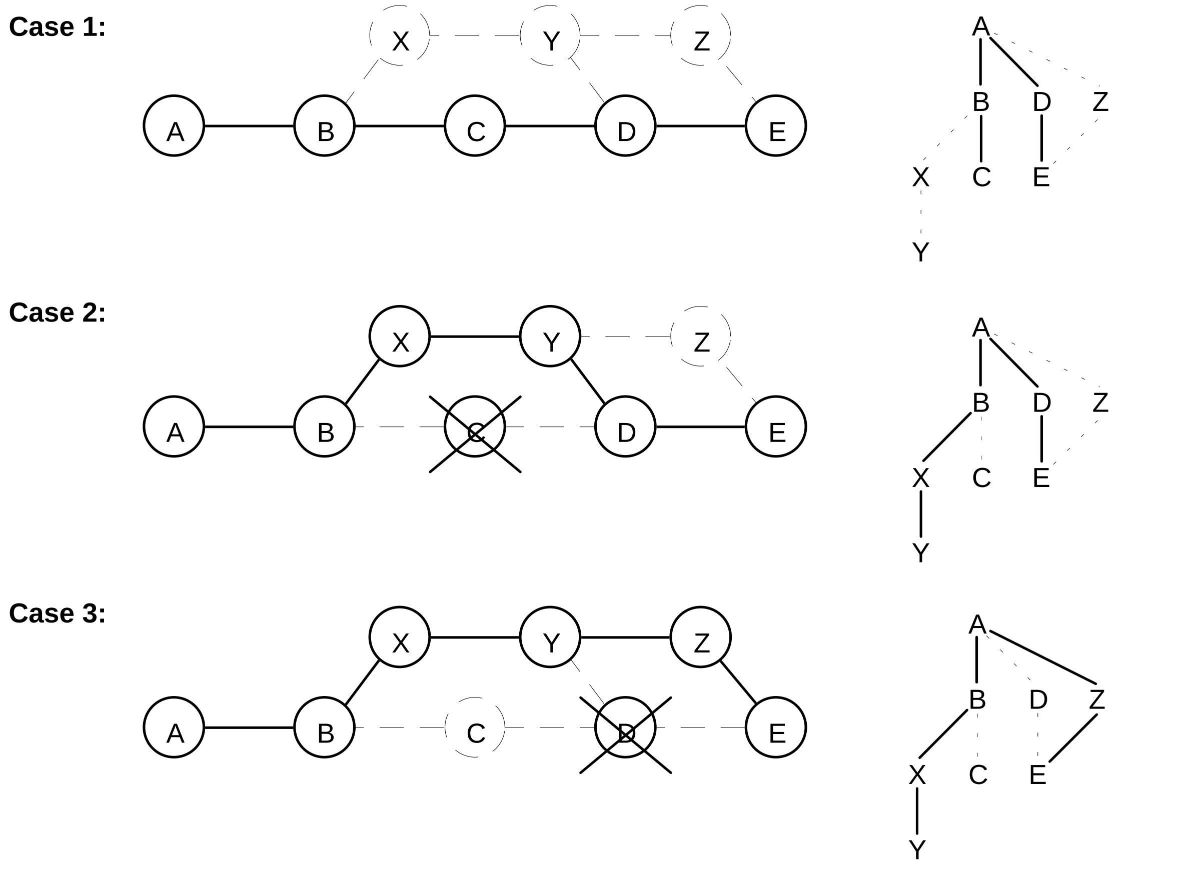
- case 1 (normal): A-B-C-D-E;
- case 2 (resulting from malfunctioning C): A-B-X-Y-D-E;
- case 3 (resulting from malfunctioning D): A-B-X-Y-Z-E.

The following X-interface relations apply (of course other X-interface relations may exist, but they are not used in these particular cases; between brackets, the X-interface relations which are no longer used are mentioned):

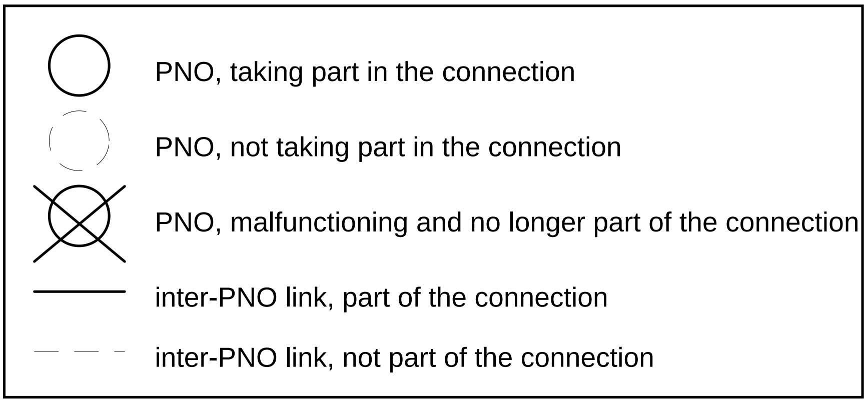
- case 1: A-B, B-C, A-D and D-E;
- case 2: A-B, (B-C), B-X, X-Y, A-D and D-E;
- case 3: A-B, (B-C), B-X, X-Y, (A-D), (D-E), A-Z and Z-E.

For these cases, the following scenarios are elaborated:

- scenario 1: PNO A establishes a VP / VC connection with E;
- scenario 2: in the end situation of scenario 1, something goes wrong with the connections around PNO C (i.e. going from case 1 to case 2);
- scenario 3: in the end situation of scenario 1, something goes wrong with the connections around PNO D (i.e. going from case 1 to case 3);
- scenario 4: in the end situation of scenario 1, the connection parameters are modified, without changing the route.



(connection maps) (X-interface relation trees)


PNO, taking part in the connection

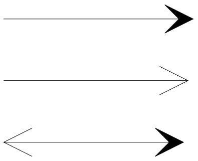
PNO, not taking part in the connection

PNO, malfunctioning and no longer part of the connection

inter-PNO link, part of the connection

inter-PNO link, not part of the connection

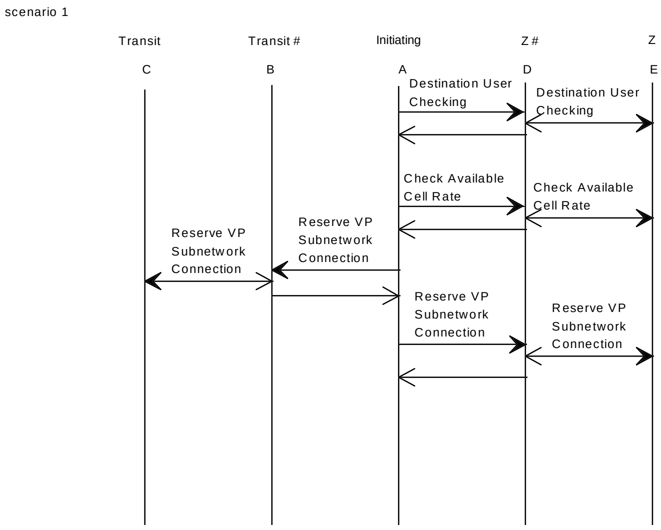
In the next figures (flow charts), the following legend is applicable:


request (named after the Management Functions)

positive answer after a request

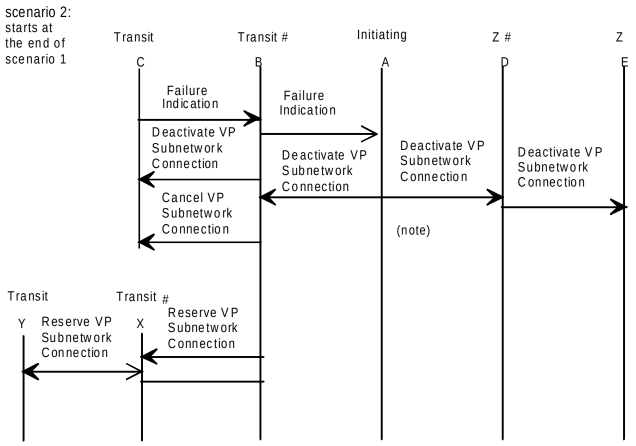
request and positive answer presented together
(in fact there is a timing difference; for reasons of
simplicity combined)

Scenario 1: PNO A establishes a VP / VC connection with E.


scenario 1
Transit
Transit #
Initiating
Z #
Z
C
B
A
D
E
Destination User Checking
Destination User Checking
Check Available Cell Rate
Check Available Cell Rate
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection

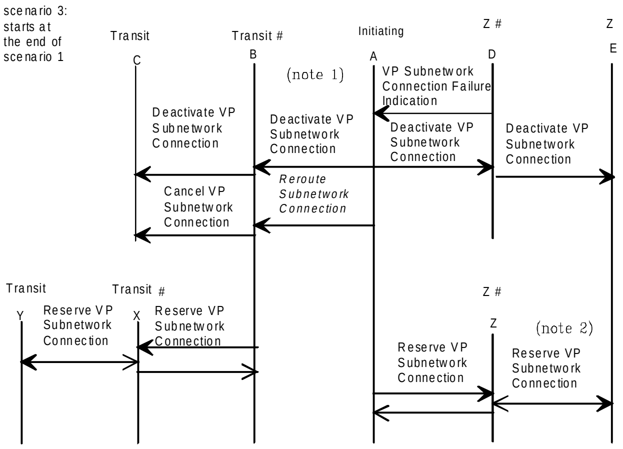
NOTE: Transit # and Z # act respectively as Transit and Z from the view point of the Initiating, even though they use other Transit and Z PNOs for the establishment of the connection.

Scenario 2: in the end situation of scenario 1, something goes wrong with the connections around PNO C (i.e. going from case 1 to case 2).


scenario 2: starts at the end of scenario 1
Transit
Transit #
Initiating
Z #
Z
C
B
A
D
E
Failure Indication
Failure Indication
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Cancel VP Subnetwork Connection
(note)
Transit
Transit #
Y
X
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection

NOTE: At this point the Initiating or the Z might decide to cancel the whole connection, then the rest of the scenario is no longer applicable.

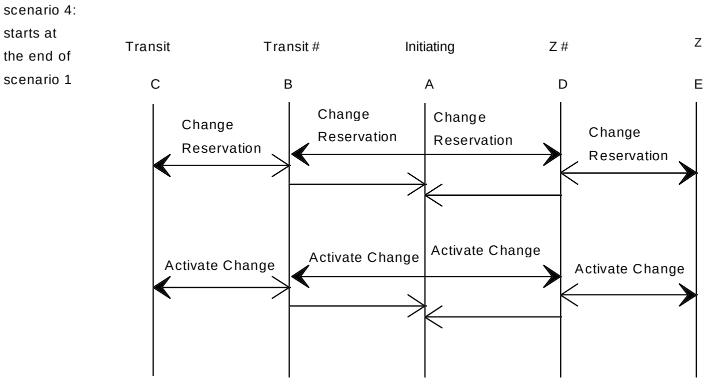
Scenario 3: in the end situation of scenario 1, something goes wrong with the connections around PNO D (i.e. going from case 1 to case 3).


scenario 3:
starts at
the end of
scenario 1
Transit
C
Transit #
B
Initiating
A
Z #
D
Z
E
(note 1)
VP Subnetwork Connection Failure Indication
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Deactivate VP Subnetwork Connection
Cancel VP Subnetwork Connection
Reroute Subnetwork Connection
Transit
Y
Transit #
X
Z #
Z
(note 2)
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection
Reserve VP Subnetwork Connection

NOTE 1: At this point the Initialing or the Z might decide to cancel the whole connection, then the rest of the scenario is no longer applicable.

NOTE 2: Since E receives a request for an already allocated VP identifier, E will recognize this as a request to reconnect to the existing connection to its destination user.

Scenario 4: in the end situation of scenario 1, the connection parameters are modified, without changing the route.



Annex E (informative): Traffic Descriptor reference problems

Problems in the Abstract Syntax Notation One (ASN.1) part of the present document.

The "Schedule" datatypes (Daily-, Weekly-, Monthly-, Occasional-Schedule) that are in VpSchedulers, are also defined in I-ETS 300 653 [3]. The reason that they are not imported from I-ETS 300 653 [3] is that in I-ETS 300 653 [3] TrafficDescriptor cannot be found.

The situation can be illustrated like this:

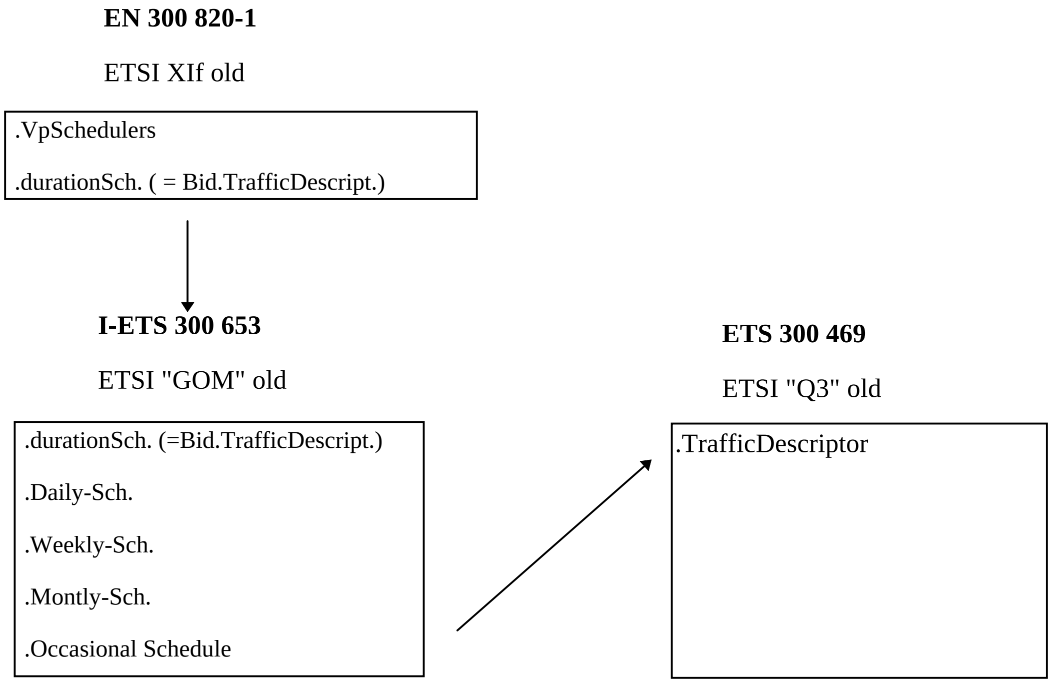


Figure E.1: trafficDescriptor reference during the development of the X-interface

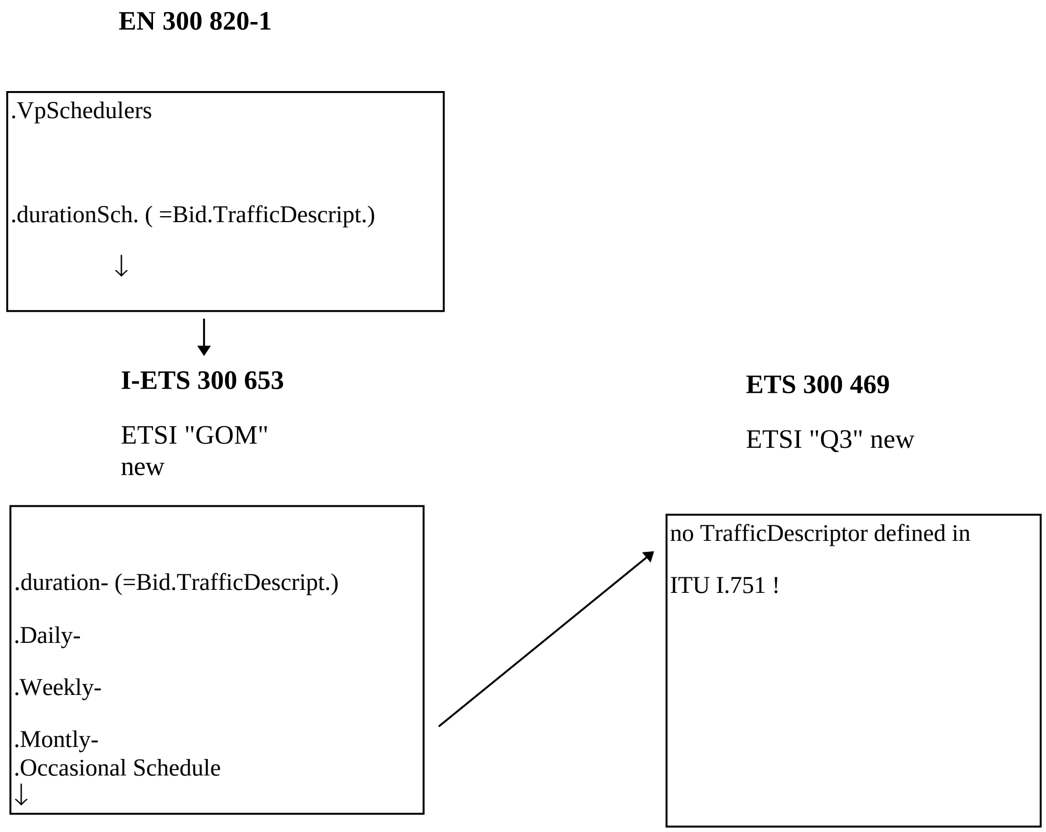


Figure E.2: trafficDescriptor reference problems

The problem is the endorsement of ETS **300 469** with ITU-T Recommendation **I.751** [5]: ETS **300 469** now is a copy of ITU-T Recommendation I.751 [5]. There is no Type *TrafficDescriptor* defined in ITU-T Recommendation **I.751** [5], so I-ETS **300 653** [3] imports a non-existing Datatype.

To cope with this the Typedefinition *TrafficDescriptor* is included in the present document and all "ScheduleTypes" (and their "subtypes") that are also in I-ETS **300 653** [3] are defined again (copied) in the present document to prevent "circular definitions".

It is recommended that the typedefinition of *TrafficDescriptor* (the one that is defined in the present document) will be included in I-ETS **300 653** [3] in order to be able to remove the typedefinition of the "ScheduleTypes" from the present document.

NOTE: There is a typedefinition *BandwidthScheduling* in the I-ETS 300 653 [3] that is the same as *VpSchedulers*, only *monthlySchedule* and *occasionalSchedule* are interchanged.

So perhaps later, it is also possible to remove the typedefinition of VpSchedulers from the present document.

Annex F (informative): Bibliography

The following material, though not specifically referenced in the body of the present document, gives supporting information.

- ITU-T Recommendation I.610: "B-ISDN operation and maintenance principles and functions".
- ITU-T Recommendation M.3010: "Principles for a Telecommunications management network".
- ITU-T Recommendation Q.822: "Stage 1, stage 2 and stage 3 description for the Q3 interface - Performance management".
- ITU-T Recommendation Q.821: "Stage 2 and stage 3 description for the Q3 interface - Alarm surveillance".
- ITU-T Recommendation X.720: "Information technology - Open Systems Interconnection - Structure of management information: Management information model".
- ITU-T Recommendation X.722: "Information technology - Open Systems Interconnection - Structure of Management Information: Guidelines for the definition of managed objects".
- ITU-T Recommendation X.208: "Specification of Abstract Syntax Notation One (ASN.1)".
- 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 G.774: "Synchronous Digital Hierarchy (SDH) management information model for the network element view".
- ETS 300 371: "Transmission and Multiplexing (TM); Plesiochronous Digital Hierarchy (PDH) information model for the Network Element (NE) view".
- ITU-T Recommendation M.3020: "TMNinterface specification methodology".
- ITU-T Recommendation I.371: "Traffic control and congestion control in B-ISDN".
- ITU-T Recommendation I.356: "B-ISDN ATM layer cell transfer performance".
- ITU-T Recommendation I.357: "B-ISDN semi-permanent connection availability".
- ITU-T Recommendation I.751: "Asynchronous transfer mode management of the network element view".
- ETS 300 469: "Broadband Integrated Services Digital Network (B-ISDN); Asynchronous Transfer Mode (ATM); Management of the network element view [ITU-T Recommendation I.751 (1996)]".
- EN 300 820-3: "Telecommunications Management Networks (TMN); Management information model for the T-type interface between Opeartion Systems (OSs) of a Virtual Path (VP) / Virtual Channel (VC) cross connected network; Part 3: Performance management aspects".

History

<table>
<tr><th colspan="3">Document history</th></tr>
<tr><td>Edition 1</td><td>December 1996</td><td>Public Enquiry as ETS 300 820-1 PE 120: 1996-12-16 to 1997-04-11</td></tr>
<tr><td>V1.1.1</td><td>April 1998</td><td>Vote V 9826: 1998-04-28 to 1998-06-26</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>