INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

I.751

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(03/96)

INTEGRATED SERVICES DIGITAL NETWORK (ISDN)

B-ISDN EQUIPMENT ASPECTS

ASYNCHRONOUS TRANSFER MODE MANAGEMENT OF THE NETWORK ELEMENT VIEW

ITU-T Recommendation I.751

(Previously “CCITT Recommendation”)

FOREWORD

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

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

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation I.751 was prepared by ITU-T Study Group 15 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 19th of March 1996.

NOTE

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

CONTENTS

			Page
1	Scope		1
	1.1	Scope of this Recommendation	1
	1.2	Structure of this Recommendation	2
2	References		2
3	Definitions		4
4	Abbreviations		5
5	ATM Network Element Management overview		7
	5.1	ATM Network Management architecture	7
	5.2	ATM NE Management ensembles	8
6	ATM Network element plane management requirements		11
	6.1	Common management requirements	11
	6.2	Cell level protocol management	15
	6.3	Virtual Path/Virtual Channel management	17
7	Management information model		21
	7.1	Information model overview	21
	7.2	Managed Object Class Definitions	21
	7.3	Conditional Packages	40
	7.4	Attributes	45
	7.5	Actions	59
	7.6	Notifications	61
	7.7	Parameter definitions	61
	7.8	Name-Bindings	61
	7.9	Supporting productions	68
Appendix I – Segment monitoring objects			72
Appendix II – Examples of relationship class and relationship mappings			75
	II.1	Characterises relationship	75
	II.2	Client to server relationship	76
	II.3	Sublayer client to server relationship	76
	II.4	Connection relationship	77
	II.5	Supporting productions	77
Appendix III – Multipoint requirements and object classes			78
	III.1	Requirements	78
	III.2	Managed object class definitions	78
	III.3	Actions	79

SUMMARY

This Recommendation provides management requirements and an information model which pertain to the plane management of the ATM network element. The information model describes the managed object classes and their properties that are used to describe the information exchanged across management interfaces defined in Recommendation M.3010 Telecommunications Management Network (TMN) architecture. This Recommendation specializes the generic object classes of Recommendations M.3100, Q.821 and Q.822 to provide the information model specific to the ATM network element.

KEYWORDS

Action, ASN.1, Asynchronous Transfer Mode, Attribute, GDMO, Information Model, Managed Object Class, Notification

Recommendation I.751

ASYNCHRONOUS TRANSFER MODE MANAGEMENT OF THE NETWORK ELEMENT VIEW

(Geneva, 1996)

1 Scope

1.1 Scope of this Recommendation

This Recommendation covers the ATM NE plane management requirements and information model supporting the ATM layer, including any management requirements for the adaptation between the physical layer and the ATM layer (transport path/VP level adaptation). Functional capabilities of the ATM NE are given in Recommendation I.731 [18], and in Recommendation I.732 [19]. The physical layer management requirements are defined in the respective technology Recommendations and only referred to here.

This Recommendation addresses the following areas of ATM network element management:

- *Fault Management*

  The Fault Management requirements covered in this Recommendation include the autonomous reporting of ATM NE component failures, physical facility failures, and ATM link/connection failures.

- *Performance Management*

  The Performance Management requirements covered in this Recommendation include:

  1) physical layer (e.g. SDH path level) performance monitoring;
  2) Transmission Convergence Level performance monitoring;
  3) ATM Layer protocol monitoring;
  4) UPC/NPC violation monitoring;
  5) Traffic monitoring; and
  6) VP/VC performance monitoring.

- *Configuration Management*

  The Configuration Management requirements covered in this Recommendation include:

  1) ATM NE configuration identification and change reporting;
  2) the configuration of UNIs and NNIs;
  3) the cross-connection of point-to-point VP and VC links[1)] and the configuration of point-to-point VPCs and VCCs;
  4) the configuration of VPC and VCC segment end-points; and
  5) management application control (e.g. suppression) of automatically generated ATM NE reports.

For the management interface, it is necessary that the management requirements and the information model described herein match the functionality supported by the ATM NE described in Recommendation I.732 [19]. Recommendation I.732 [19] does not require all functionality to be present in all ATM NEs. Therefore, not all the management requirements and all parts information model described in this Recommendation need to be present in a given ATM NE. The management interface requirements are independent of the NE implementation, either technological or geographical.

---

1) Note that this Recommendation uses the term “link” from Recommendation I.321 [12], rather than the term “link connection” from Recommendation I.326 [13].

1.2 Structure of this Recommendation

This Recommendation is organized as follows:

- Clauses 2, 3, and 4 provide respectively for references, definitions and abbreviations relevant to this Recommendation.
- Clause 5 provides an overview of the management of the ATM network. It succinctly describes the ATM network management architectures and the ATM network element management ensembles.
- Clause 6 describes the ATM network element plane management requirements.
- Clause 7 describes the information model using the notation mechanisms defined in Recommendation X.722 [38], the relationships among managed object classes, and the syntax definitions of the information carried in the protocol using Abstract Syntax Notation One (ASN.1) defined in Recommendation X.208 [32].

2 References

The following Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other reference are subject to revision: all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

[1] ITU-T Recommendation G.707 (1996), *Network node interface for the Synchronous Digital Hierarchy (SDH).*

[2] ITU-T Recommendation G.773 (1993), *Protocol suites for Q-interfaces for management of transmission systems*.

[3] CCITT Recommendation G.774 (1992), *Synchronous Digital Hierarchy (SDH) management information model for the network element view.*

[4] ITU-T Recommendation G.774.1 (1994), *Synchronous Digital Hierarchy (SDH) performance monitoring for the network element view.*

[5] ITU-T Recommendation G.774.2 (1994), *Synchronous Digital Hierarchy (SDH) configuration of the payload structure for the network element view*.

[6] ITU-T Recommendation G.774.3 (1994), *Synchronous Digital Hierarchy (SDH) management of multiplex-section for the network element view*.

[7] ITU-T Recommendation G.774.4 (1995), *Synchronous Digital Hierarchy (SDH) management of the subnetwork connection protection for the network element view*.

[8] ITU-T Recommendation G.774.5 (1995), *Synchronous Digital Hierarchy (SDH) management of connection supervision functionality (HCS/LCS) for the network element view.*

[9] ITU-T Recommendation G.805 (1995), *Generic functional architecture of transport networks*.

[10] ITU-T Recommendation G.826 (1993), *Error performance parameters and objectives for international, constant bit rate digital paths at or above the primary rate.*

[11] ITU-T Recommendation I.311 (1993), *B-ISDN general network aspects*.

[12] CCITT Recommendation I.321 (1991), *B-ISDN protocol reference model and its application.*

[13] ITU-T Recommendation I.326 (1995), *Functional architecture of transport networks based on ATM*.

[14] ITU-T Recommendation I.353 (1993), *Reference events for defining ISDN performance parameters*.

[15] ITU-T Recommendation I.356 (1993), *B-ISDN ATM layer cell transfer performance*.

[16] ITU-T Recommendation I.371 (1993), *Traffic control and congestion control in B-ISDN*.

[17] ITU-T Recommendation I.610 (1995), *B-ISDN operation and maintenance principles and functions*.

[18] ITU-T Recommendation I.731 (1996), *Types and general characteristics of ATM equipment*.

[19] ITU-T Recommendation I.732 (1996), *Functional characteristics of ATM equipment*.

[20] CCITT Recommendation M.20 (1992), *Maintenance philosophy for telecommunications networks*.

[21] ITU-T Recommendation M.60 (1993), *Maintenance terminology and definitions*.

[22] ITU-T Recommendation M.2100 (1995), *Performance limits for bringing-into-service and maintenance of international PDH paths, sections and transmission system.*

[23] CCITT Recommendation M.2120 (1992), *Digital path, section and transmission system fault detection and localization procedures.*

[24] CCITT Recommendation M.3010 (1992), *Principles for a telecommunications management network.*

[25] ITU-T Recommendation M.3100 (1995), *Generic network information model.*

[26] CCITT Recommendation M.3400 (1992), *TMN management functions*.

[27] ITU-T Recommendation M.3610 (1996), *Principles for applying the TMN concept to the management of B-ISDN.*

[28] ITU-T Recommendation Q.811 (1993), *Lower layers protocols profiles for the Q3 interface*.

[29] ITU-T Recommendation Q.812 (1993) – *Upper layer protocol profiles for the Q3 interface*.

[30] ITU-T Recommendation Q.821 (1993), *Stage 2 and stage 3 description for the Q3 interface – alarm surveillance*.

[31] ITU-T Recommendation Q.822 (1994), *Stage 1, stage 2 and stage 3 description for the Q3 interface – Performance management*.

[32] CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1)*.

[33] CCITT Recommendation X.701 (1992), *Information technology – Open Systems Interconnection – Systems management overview.*

[34] CCITT Recommendation X.710 (1991), *Common management information service definitions for CCITT applications.*

[35] CCITT Applications. Recommendation X.711 (1991), *Common management information protocol specification for CCITT applications.*

[36] CCITT Applications. Recommendation X.720 (1992), *Information technology – Open Systems Interconnection – Structure of management information: Management information model.*

[37] CCITT Recommendation X.721 (1992), *Information technology – Open Systems Interconnection – Structure of management information: Definition of management information.*

[38] CCITT Recommendation X.722 (1992), *Information technology – Open Systems Interconnection – Structure of management information: Guidelines for the definition of managed objects.*

[39] ITU-T Recommendation X.723 (1993), *Information technology – Open Systems Interconnection – Structure of management information: Generic management information.*

[40] ITU-T Recommendation X.725 (1995), *Information technology – Open Systems Interconnection – Structure of management information: General relationship model.*

[41] CCITT Recommendation X.730 (1992), *Information technology – Open Systems Interconnection – Systems Management: Object management function.*

[42] CCITT Recommendation X.731 (1992), *Information technology – Open Systems Interconnection – Systems Management – State management function.*

[43] CCITT Recommendation X.733 (1992), *Information technology – Open Systems Interconnection – Systems Management: Alarm reporting function.*

[44] CCITT Recommendation X.734 (1992), *Information technology – Open Systems Interconnection – Systems Management: Event report management function.*

[45] CCITT Recommendation X.735 (1992), *Information technology – Open Systems Interconnection – Systems Management: Log control function.*

[46] CCITT Recommendation X.736 (1992), *Information technology – Open Systems Interconnection – Systems Management: Security alarm report function.*

[47] ITU-T Recommendation X.738 (1993), *Information technology – Open Systems Interconnection – Systems Management: Summarization function.*

[48] ITU-T Recommendation X.739 (1993), *Information technology – Open Systems Interconnection – Systems Management: Metric objects and attributes.*

[49] CCITT Recommendation X.740 (1992), *Information technology – Open Systems Interconnection – Systems Management: Security audit trail function.*

[50] ITU-T Recommendation X.741 (1995), *Information technology – Open Systems Interconnection – Systems Management: Objects and attributes for access control.*

[51] ITU-T Recommendation X.744[2)], *Information technology – Open Systems Interconnection – Systems Management – Software management function.*

[52] ITU-T Recommendation X.745 (1993), *Information technology – Open Systems Interconnection – Systems Management –Test management function.*

[53] ITU-T Recommendation X.749[2)], *Systems Management – Management domain – Management function and Management policy management function.*

3 Definitions

For the purposes of this Recommendation, the following definitions apply.

3.1 **agent**: Part of the MAF which is capable of responding to network management operations issued by a manager and may perform operations on managed objects, issuing events on behalf of managed objects. The managed objects can reside within the entity or in another open system. Managed objects from other open systems are controlled by a distant agent via a local manager. All ATM NEs will support at least one agent. Some ATM NEs will provide managers and agents (being managed).

3.2 **management application function (MAF)**: An application process participating in system management. The management application function includes an agent (being managed) and/or manager. Each ATM network element (NE) and operations system or mediation device (OS/MD) must support a management application function that includes at least one agent. A management application function is the origin and termination for all TMN messages.

3.3 **manager**: Part of the MAF which is capable of issuing network management operations (i.e. retrieve alarm records, set thresholds) and receiving events (i.e. alarms, performance). ATM NEs may or may not include a manager while ATM OS/MDs will include at least one manager.

3.4 **message communications function (MCF)**: The message communications function provides facilities for the transport of TMN messages to and from the MAF, as well as facilities for the transit of messages. The message communications function does not originate or terminate messages (in the sense of the upper protocol layers).

3.5 **managed object (MO)**: The management view of a resource within the telecommunication environment that may be managed via the agent. Examples of ATM managed objects are: equipment, receive port, transmit port, power supply, plug-in card, virtual container, and multiplex section.

---

2) Presently at the stage of draft.

3.6 managed object class (MOC): An identified family of managed objects that share the same characteristics, e.g. “equipment” may share the same characteristics as “plug-in card”.

3.7 network element (NE): A set of equipment or TMN entities within the telecommunication network that performs at least NEFs and may also support OSF/MFs. It contains managed objects, a MCF and a MAF. A network element contains equipment that may or may not be geographically distributed.

3.8 network element function (NEF): A function within an ATM entity that supports the ATM-based network transport services, e.g. multiplexing, cross-connection. The network element function is modelled by managed objects.

3.9 intra-network node interface (intra-NNI): An interface between two nodes within the network of a single Public Network Operator.

3.10 inter-network node interface (inter-NNI): An interface between two nodes, each belonging to the networks of two different Public Operators.

3.11 operations system function or mediation function (OSF/MF): A Telecommunications Management Network (TMN) entity that processes management information to monitor and control the ATM network. In the ATM sub-portion of the TMN, no distinction is made between the operations system function and the mediation function; this entity being a MAF containing at least one manager.

3.12 operations system or mediation device (OS/MD): A stand-alone physical entity that supports OSF/MFs but does not support NEFs. It contains a Message Communications Function (MCF) and a MAF.

3.13 segment: An arbitrary series of contiguous links for management purposes, typically under the control of one Administration or organization. However, it can be extended beyond the control of one Administration/organization by mutual agreement.

3.14 user network interface (UNI): An interface between two nodes, one belonging to a user network, the other to a Public Operator.

3.15 UPC/NPC: A set of actions taken by the ATM NE to monitor and control traffic. Their main purpose is to detect violations of negotiated traffic parameters and to take appropriate action. These functions are described in Recommendation I.732 [19].

4 Abbreviations

For the purposes of this Recommendation, the following abbreviations are used.

AIS	Alarm Indication Signal
CDV	Cell Delay Variation
CLP	Cell Loss Priority
CTP	Connection Termination Point
DCP	Downstream Connectivity Pointer
HEC	Header Error Check
ISO	International Organization for Standardization
Inter-NNI	inter-Network Node Interface
Intra-NNI	intra-Network Node Interface
LCD	Loss of Cell Delineation
MAF	Management Application Function
MCF	Message Communications Function

MD	Mediation Device
MF	Mediation Function
MO	Managed Object
MOC	Managed Object Class
MS	Multiplex Section
NE	Network Element
NEF	Network Element Function
NEL	Network Element Layer
NEML	Network Element Management Layer
NML	Network Management Layer
NPC	Network Parameter Control
OAM	Operations, Administration and Maintenance
OS	Operations System
OSF	Operations System Function
OSI	Open Systems Interconnection
PCR	Peak Cell Rate
Pkg	Package
PM	Performance Monitoring
PT	Payload Type
PTI	Payload Type Identifier
PVC	Permanent Virtual Connection
QOS	Quality of Service
RDI	Remote Defect Indication
RDN	Relative Distinguished Name
SDH	Synchronous Digital Hierarchy
SCR	Sustained Cell Rate
STM-N	Synchronous Transport Module N
SVC	Switched Virtual Circuit
TC	Transmission Convergence
TMN	Telecommunications Management Network
TTP	Trail Termination Point
TP	Termination Point
UNI	User Network Interface
UCP	Upstream Connectivity Pointer
UPC	Usage Parameter Control
VP	Virtual Path
VC	Virtual Channel
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VCL	Virtual Channel Link
VPC	Virtual Path Connection
VPI	Virtual Path Identifier
VPL	Virtual Path Link

5 ATM Network Element Management overview

5.1 ATM Network Management architecture

The Telecommunications Management Network (TMN) model defined in Recommendation M.3010 [24] partitions network management functions into five hierarchical layers: Business Management Layer, Service Management Layer, Network Management Layer (NML), Network Element Management Layer (NEML), and Network Element Layer (NEL). Each layer provides the view required by the next higher layer to perform its functions. The TMN Model presented in Recommendation M.3010 [24] defines reference points between each layer that may or may not be implemented as physical interfaces. The reference points of interest in this Recommendation are the q3 logical reference points used to manage ATM NEs. These logical interfaces, at the ATM NE level, are shown in Figure 5-1. The TMN Model may be realized in a number of different ways. Representative examples of four alternative physical realizations are shown in Figures 5-2 a), b), c) and d). Notice from Figure 5-2, that realizations of the q3 reference point are labelled Q3 Interfaces.

The Q3 Interface requirements defined in this Recommendation focus on the Network Element Management Layer (NEML) to Network Element Layer (NEL) and Network Management Layer (NML) to Network Element Management Layer (NEML) interactions needed to support ATM NE management, where an ATM NE may be realized as either a stand-alone device or geographically distributed system. With respect to Figure 5-2, the requirements defined in this Recommendation are relevant to ATM NEs supporting either NEL capabilities or a combination of NEL and NEML capabilities as well as to the operations support system(s) that manage them.

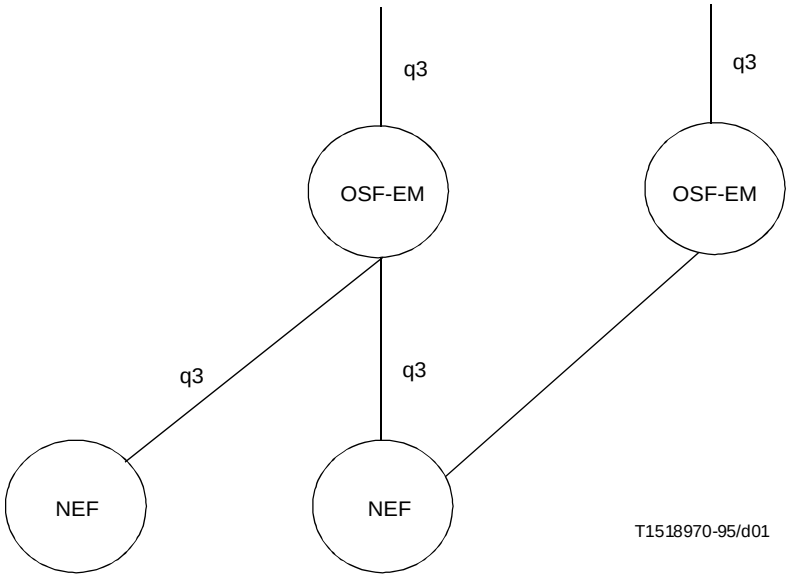
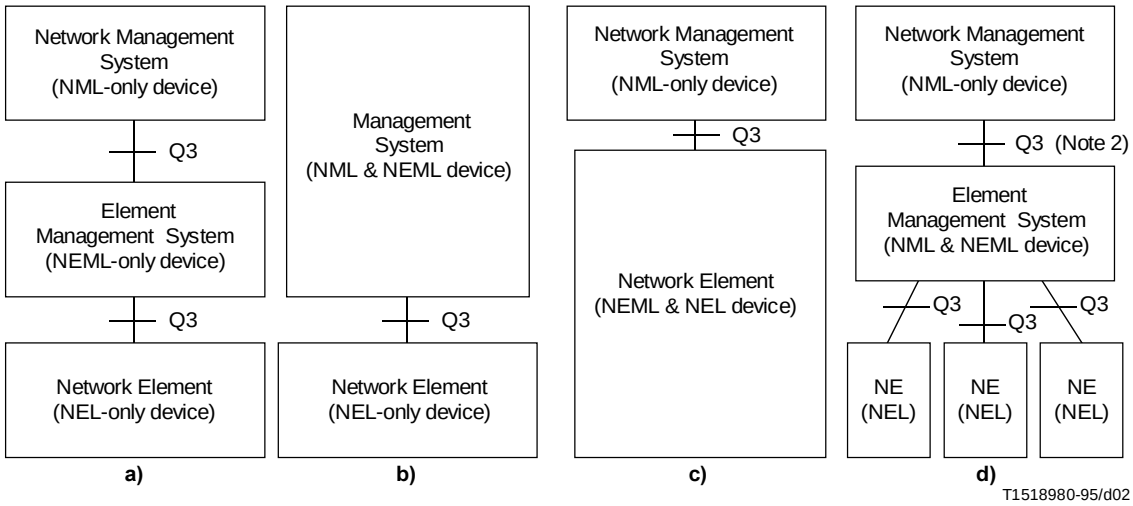


FIGURE 5-1/I.751

Example of Logical TMN Architecture



NOTES

1 The Q3 interfaces depicted in the figure show where the object classes defined in this Recommendation could apply.

2 The Q3 interface defined in this Recommendation is not applicable for the NML- to NML-interactions, however it is applicable for the NEML- to NML-interactions.

FIGURE 5-2/I.171

Physical realization examples of the TMN model (Note 1)

5.2 ATM NE Management ensembles

This subclause provides an overview of the management of a Transport Network Element. The management of a transport Network Element can be divided into a set of individual management ensembles where each ensemble addresses a particular management objective.

This management overview should help structuring the specification of management of transport equipment into a set of smaller well-defined ensembles and guide readers of TMN specifications to find out the various management aspects addressed in TMN specifications and those aspects which are relevant for the management of a transport Network Element.

Some ensembles are specific to a transport technology and others are independent of any transport technology and are common to all types of transport NEs. An ensemble is a collection of objects which altogether perform a given management objective, thus an ensemble could cover several functional area such as configuration and fault management. The following provides a list of ensembles and, in addition to the definition of ensembles, it identifies the list of ITU-T Recommendations covering a particular ensemble.

5.2.1 Transport technology specific ensembles

ATM

- ATM Layer Management: This ensemble covers the management of transport path/VP adaptation, Virtual Path layer, and Virtual Channel layer. It addresses the configuration and fault management of these layers. It covers the configuration of a particular ATM interface, the allocation of bandwidth and VPI/VCI range attached to that particular interface. It also covers the detection and reporting of alarms which pertain to these layers.

  Related Recommendation: I.751.

- ATM VP/VC Connections Management: This ensemble covers the management (configuration and fault) of Virtual Path and Virtual Channel connections in an ATM network element. This ensemble covers the set-up and release of connections in ATM equipment, as well as the allocation of VPI or VCI for these connections and the activation and deactivation of cell flow attached to the connection.

  Related Recommendations: I.751, M.3100 [25].

- ATM Performance Management: This ensemble covers the monitoring of performance parameters (gauge and counters) for the transport path/VP adaptation, and for the Virtual Path and Virtual Channel layers.

  Related ITU Recommendations: I.751, Q.822 [31].

SDH

- SDH Transport Management: This ensemble covers the management of SDH transmission media layers and of higher and lower order path layers. It addresses the configuration and fault management of SDH transport resources defined by the following G.783 functional blocks.

  Related Recommendations: G.774 [3], G.774.2 [5], G.774.3 [6], G.774.5 [8].

- SDH Connection Management: This ensemble covers the management (configuration and fault) of higher order and lower order connections and of sub-network connection protection (SNCP).

  Related Recommendations: G.774.4 [7], M.3100 [25].

- SDH Synchronization: This ensemble covers the management (configuration and fault) of the generation of timing reference points.

  Related Recommendations: no ITU-T Recommendation currently available.

- SDH Performance Management: This ensemble covers the monitoring of performance parameters (gauge and counters) for all SDH layer networks according to Recommendation G.784 and the monitoring of protection switch events related to APS.

  Related Recommendations: G.774.1 [4], Q.822 [31].

- SDH Overhead Management: This ensemble covers the management (configuration and fault) of SDH overhead handling.

  Related Recommendations: G.774 [3], M.3100 [25].

PDH

- PDH Transmission Management: This ensemble covers the management (configuration and fault) of PDH layers.

  Related Recommendations: no ITU-T Recommendation currently available.

5.2.2 Transport technology independent ensembles

- Alarm Management: This ensemble covers the reporting and filtering of alarms.

  Related Recommendations: Q.821 [30], X.733 [43].

- Equipment Management: This ensemble covers the management (configuration and fault) of physical components (e.g. rack), equipment protection, and the NE itself.

  Related Recommendation: M.3100 [25].

- Software Management: This ensemble covers the management (configuration and fault) of software. It typically includes control of software downloading, activation, backup/restore.

  Related Recommendations: M.3100 [25], X.744 [51].

- TMN Communication Management: This ensemble covers the management (configuration, fault and performance) of the whole TMN communication protocol stack. It addresses management of directory (Application Entity) and routing information.

  Related ITU Recommendation: X.723 [39].

- Support Management: This ensemble covers the management (configuration and fault) of support functions such as logging, forwarding, summarization facilities, etc.

  Related Recommendations: X.721 [37], X.733 [43], X.734 [44], X.735 [45], X.738 [47], X.739 [48], Q.821 [30].

- Security Management: This ensemble covers the management of authentication and access control when accessing managed objects.

  Related Recommendations: X.736 [46], X.740 [49], X.741 [50], X.749 [53].

See Fig. 5-3.

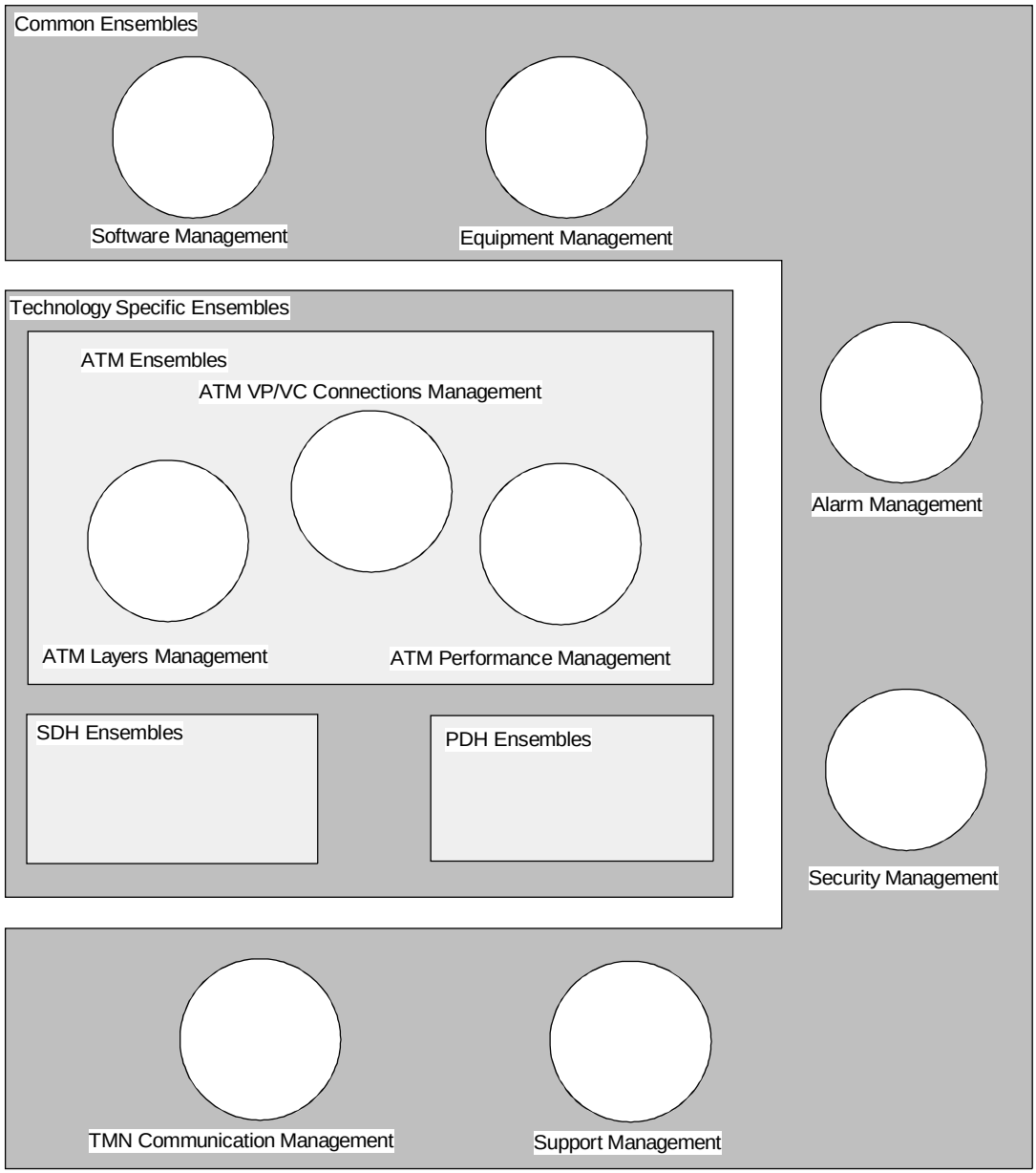


FIGURE 5-3/I.751

Transport NE management ensembles

6 ATM Network element plane management requirements

This clause provides element-level management requirements placed upon the operation interface to an ATM network element.

6.1 Common management requirements

This subclause covers the management requirements which are common to the transport path/VP adaptation, the virtual path level, and the virtual channel level.

6.1.1 Fault management

Fault management is concerned with the detection and reporting of relevant events and conditions which occur in the network. Alarms are indications that are automatically generated by an NE as a result of certain events and conditions. The OS shall have the ability to define which events and conditions generate autonomous alarm reports, and which shall be reported on request.

The following alarm-related requirements shall be supported:

6.1.1.1 Inhibition and resumption of defect detection

Inhibition and resumption of defect detection allows the OS to control the detection of specified type of alarm on a specified set of TP. The inhibited defect will never be detected and therefore not reported to the management. The presence of the defect is not visible at the management interface.

6.1.1.2 Fault persistency

Fault persistency allows the NE to wait a certain amount of time before to potentially generate alarms, this temporal filter depends on the defect appearance characteristic and frequency (toggling, stable, ...). Only a defect which passes that filter could be reported as an alarm.

The fault persistency mechanism requires timers. These timers are defined in 6.2/I.732 [19], along with their default values. The Network Element should be designed to handle a range of values for the timers, depending on the operator's practices. However, the setting of the timers takes place only once, at the installation of the Network Element. Therefore, it is not required to modify the timers over the operations interface. They may be modified at installation time.

6.1.1.3 Service dependency

Service dependency allows the NE to assess the impact of the failure on the resource from the service view point. In the context of this Recommendation, the service means a service provided by the transfer plane. The service is considered not affected if a failure has been recovered using protection switch residing within the network element.

6.1.1.4 Correlation of Alarm

Correlation of Alarm is used to allow the NE to report only the root cause of the event. It may indicate that an alarm was a consequent (secondary) of another alarm (primary), this information being useful for fault localization and also alarm filtering, since secondary alarm may be filtered by the alarm logging and the alarm reporting functions providing they have been correlated. Correlation between a Loss Of Cell Delineation and the consequent VP AIS alarms, or correlation between an equipment alarm and the alarm sent by the termination point which has been affected by the equipment failure, are examples of possible alarm correlation.

6.1.1.5 Severity assignment

Severity assignment is used to associate a perceived severity to the alarm which may be generated by the NE. The severity of the alarm depends on the management application. As an example, the severity of an alarm which could depend on the service impact of the event, a non-service affecting event may be reported as a Warning or Minor alarm, while service affecting events may be reported as Major or Critical.

The Operations interface shall support OS requests to assign a severity to each alarm generated by each externally managed physical component of the ATM NE.

6.1.1.6 Alarm Filtering

Alarm Filtering is used to filter the alarm notification depending on the contents of the alarm such as the type and cause of the alarm, the source of the alarm, the severity, the correlation information prior to the reporting and/or logging of alarms.

The operations interface shall support messages used to configure the ATM NE Filter. The OS shall be able to configure the ATM NE such that notifications are suppressed based on one or more of the following:

- Notification Type (e.g. specific alarm or threshold crossing alert).
- Specific Aspect(s) of a Notification Type (e.g. perceived severity).
- Type of Managed Entity Reporting the Notification.
- Specific Aspect(s) of the Managed Entity Reporting the Notification.

6.1.1.7 Alarm logging and alarm reporting

Alarm Reporting functions allows the reporting of those alarms which have passed successfully alarm filtering. Alarms are then sent to one or more destinations.

The operations interface shall support notifications used to report failures detected by ATM NEs. Each alarm shall include the following information:

- *The failed entities or list of potentially failed entities* (if known by the ATM NE)

  Entities identified should represent the smallest replaceable/repairable unit(s) of hardware or software.

- *Trouble Description*

- *Specific Problems* (optionally provided)

  This parameter identifies further refinements (e.g. sub-cause indicator information) to the generic trouble description of the alarm.

- *Severity* (i.e. critical, major, minor, warning, indeterminate, and cleared)

  This parameter indicates the severity of the alarm. Alarm severities can be assigned by the management application only for equipment alarms and physical layer communications alarms.

- *Back up Status* (optionally provided)

  This parameter indicates whether or not the entity emitting the alarm has been backed up. A value of "true" indicates that the entity has been backed up; a value of "false" indicates that the entity has not been backed up.

- *Back-up Entity* (optionally provided)

  This parameter identifies the managed entity that is providing back up services to the failed managed entity. If no back up service is being provided, the value of this parameter shall be NULL.

- *Additional Text* (optionally provided)

  This parameter is used to allow for additional text to be supplied with the alarm. Such text may further describe problem and/or failed entity (e.g. name and location).

- *Proposed Repair Actions* (optionally provided)

  This parameter, when present, is used if the cause of the alarm is known and the ATM NE can suggest one or more solutions.

- *Time and Date the failure was detected*

All the above mechanisms are shown in Figure 6-1. When all the above mechanisms are correctly assembled and configured, it is possible to control which alarms are generated in which case and for which OS.

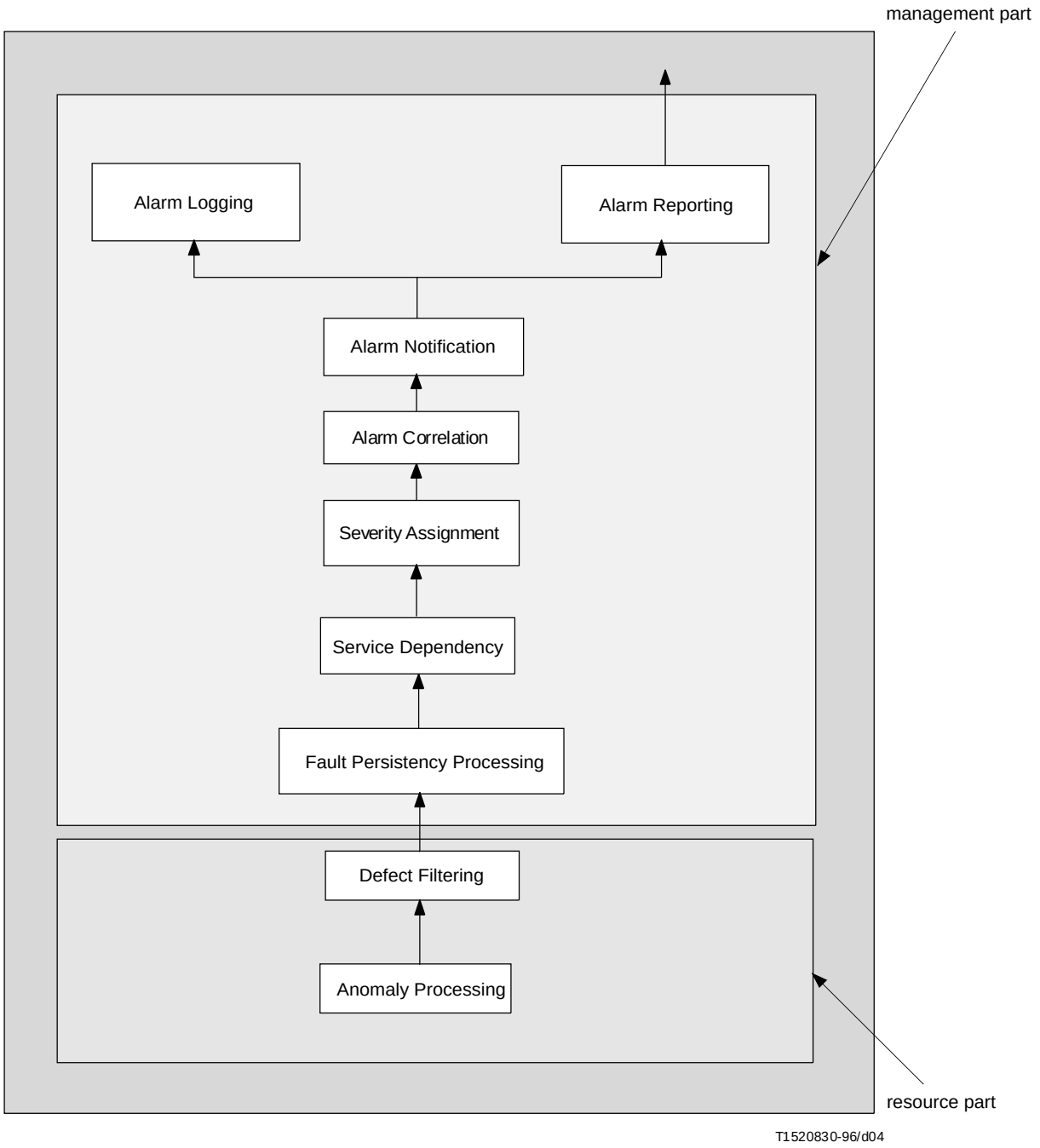


FIGURE 6-1/I.751

Functional architecture of the alarm process [3)]

6.1.2 Performance Monitoring

6.1.2.1 Performance information collection

Performance information collection refers to the ability for the NE to collect the various Performance Monitoring (PM) parameters related to a single monitored entity in that NE.

3) This sequence of mechanisms is provided only as an example in this figure.

The ATM NE shall count and store in a current PM bin the layer performance parameters identified in 6.2.2. and 6.3.2.

The operations interface shall provide the OS with the ability to reset to zero each current count for the layer performance parameters identified below.

6.1.2.2 Performance information storage requirements

Performance information storage refers to the optional capability for the NE to store PM history information on each monitored entity for prescribed time duration. The NE can also store summarized or statistical information derived from various monitored entities.

Failures, testing routines, non-boundary resets, and reconfigurations may affect the collection of performance monitoring information. When such events occur, the ATM NE is expected to flag the collected information as "suspect". The operations interface shall provide to the OS the ability to retrieve an indication as to whether the counts are reliable or suspect.

The ATM NE shall support the suppression of all-zero performance monitoring counts.

6.1.2.3 Performance management thresholding requirements

Performance management thresholding refers to the ability for the NE to inform the OS of any threshold crossing. It also provides the OS with the means for establishing thresholding criteria. When this capability is available, the following specific requirements are associated with the thresholding activity:

6.1.2.3.1 Threshold setting

The operations interface shall be support requests by the OS to retrieve and change the threshold of each register.

6.1.2.3.2 Threshold crossing reporting

As soon as a threshold is reached or crossed for a given performance event, a threshold crossing notification is generated.

6.1.2.4 Performance management reporting requirements

PM information reporting refers to the optional capability for the NE to report PM information on a scheduled basis, or as a result of a spontaneous request from the OS. A report may contain information from a given monitored entity, or it can contain summarized information or information derived statistically from a set of monitored entities. The following specific requirements are associated with the reporting activity:

6.1.2.4.1 Access to the performance information

Performance information shall be reportable when requested by the OS.

6.1.2.4.2 Periodical report of performance information

Performance information collection may be performed periodically to support trend analysis to predict future failure or degraded conditions. On request by the OS, performance information of specific ports should be reportable periodically.

6.1.3 Configuration management

6.1.3.1 Configuration change reporting

The operations interface shall support ATM NE notifications that report when the ATM NE has been installed/initialized and is now available to the OS for subsequent provisioning. These notifications need only provide a simple indication that the installation/initialization of the ATM NE is complete. Information regarding the detailed configuration of the newly installed/initialized ATM NE need not be included as part of this notification.

The operations interface shall support OS requests for information that describes the current configuration of the ATM NE.

The operations interface shall support autonomous notifications that reflect recent changes in the configuration of the ATM NE, including those that were not directly initiated by the OS over the operations interface.

The operations interface shall support notifications that reflect changes in the operational state of the various managed entities within the ATM NE. When possible, only the root operational state change shall be reported if there is no ambiguity for the impacted resources. Operational state changes that may be derived from the root state change should not be reported.

Note that this Recommendation adopts the OSI state management model defined in ITU-T Recommendation X.731 [42].

6.2 Cell level protocol management

This subclause deals with the transmission path/VP-level adaptation and with any entity common to both the VP and the VC levels.

6.2.1 Fault management

The ATM NE shall report at this layer Loss Of Cell Delineation.

6.2.2 Performance information collection

6.2.2.1 ATM Cell Level protocol performance information collection

Cell Level protocol monitoring involves collecting and thresholding information counts that measure an ATM NE's ability to successfully process and deliver incoming ATM cells. Cell Level protocol monitoring is particularly concerned with protocol abnormalities detected at the adaptation between the transmission path and the VP layers, and at the adaptation between the VP and the VC layers.

The operations interface shall provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information from each ATM interface terminating on the ATM NE:

- *Cells discarded due to HEC violations*

  This parameter provides a count of the number of incoming ATM cells discarded due to an invalid header pattern.

- *Errored Cells*

  This parameter provides a count of the number of cells which has an errored header pattern, whether or not the error was corrected.

- *Discarded Cells due to Protocol Errors*

  This parameter provides a count of the number of ATM cells misrouted due to an unrecognizable cell header field value [i.e. invalid header pattern (see 5.3.1.9/I.732 [19]), not assigned VPI/VCI value (see 5.3.1.9/I.732 [19]), out-of-range VPI/VCI value (see 5.7.1.1/I.732 [19])].

The operations interface shall provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the number of OAM cells that have entered the ATM NE per interface.

In addition, for each UNI, or NNI terminating on the ATM NE, the ATM NE should maintain one record of the latest occurrence of an ATM cell discard due to a not assigned VPI/VCI value or an out-of-range VPI/VCI value. The record shall contain:

- *The Abnormality Type*

  This attribute identifies the type of protocol error that resulted in the discarding of the ATM cell. Valid values are:

  1) not assigned VPI/VCI value; and

  2) out-of-range VPI/VCI value.

- *VPI/VCI Value of discarded cell*

- *Time and Date*

6.2.2.2 ATM traffic load information collection

The operations interface shall provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information for selected ATM interfaces terminating on the ATM NE:

- Ingress cells (whole stream).
- Egress cells (whole stream).

As an option, the operations interface should provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information for selected ATM interfaces terminating on the ATM NE:

- Ingress cells (oam cells only).
- Egress cells (oam cells only).

These counts allow the OS to assess the network utilization in real-time and allow for capacity planning.

6.2.3 Configuration management

6.2.3.1 Interface configuration

The operation interface shall support management application requests to configure the interface as an User Network Interface (UNI), an inter-Network Node Interface (inter-NNI), or an intra-Node Network Interface (intra-NNI). The following information shall be provided with each configuration request:

- *Interface ID*
- *Maximum Number of Simultaneously Active VPCs Supported*

  This parameter identifies the maximum number of VPCs that may be active across the UNI, or NNI at any one point in time.

- *Maximum Number of Simultaneously Active VCCs Supported*

  This parameter identifies the maximum number of VCCs that may be active across the UNI, or NNI at any one point in time. This information may also be provided at the VP-level.

- *Number of Allocated VPI Bits*

  This parameter identifies the number of allocated bits of the VPI sub-field for the UNI, or NNI. The value of this parameter is determined based on the lower supported value by the equipment on each end of the interface. This parameter is used by the ATM NE to select appropriate VPI values when establishing ATM connections.

- *Number of Allocated VCI Bits*

  This parameter identifies the number of allocated bits of the VCI sub-field for the UNI, or NNI. The value of this parameter is determined based on the lower supported value by the equipment on each end of the interface. This parameter is used by the ATM NE to select appropriate VCI values when establishing ATM connections. This information may also be provided at the VP-level.

The operations interface shall support network management application requests to retrieve configuration parameters associated with each UNI, or NNI terminating on the ATM NE.

The operations interface shall support network management application requests to reconfigure parameters associated with each UNI, or NNI terminating on the ATM NE.

The operations interface is capable of turning on/off the UPC/NPC function (see 5.4.1.2/I.732 and 5.8.1.2/I.732 [19]). This capability is not yet supported in the information model.

6.2.3.2 Bandwidth configuration

The operation interface should support OS requests to configure the allocation of the maximum ingress and egress transmission bandwidth at the ATM NE interface.

6.3 Virtual Path/Virtual Channel management

This subclause addresses the specific management requirements which pertains to the Virtual Path Level.

6.3.1 Fault management

6.3.1.1 Event reporting

The ATM NE shall report at this layer VP/VC AIS and VP/VC RDI at both end-points and intermediate points.

6.3.1.2 VP/VC Continuity Check

The operations interface should support optional management application requests to perform OAM Cell Continuity checks (see 6.2.1.1.2/I.610 for the VP-level continuity check, and 6.2.2.1.2/I.610 for the VC-level continuity check). The check may be continuous. An OAM Cell Continuity check is performed (by the ATM NE) by inserting/extracting a Continuity check OAM cell. The VPC/VCC OAM Cell Continuity Check monitoring are applicable to segment connection or end-to-end connection. The following information shall be supplied with each management application request to perform an OAM Cell Continuity Check:

- The Continuity Check OAM Cell Insertion Point (VP/VC CTP; VP/VC TTP).
- The Continuity Check OAM Cell Extraction Point (VP/VC CTP; VP/VC TTP).
- The flow direction OAM Cell Continuity Check indication.
- The activation/deactivation of OAM Cell Continuity Check.

The continuity check may result in the autonomous reporting of loss of continuity.

6.3.1.3 OAM Cell Loopback

The operations interface should support optional management application requests to perform synchronous OAM Cell Loopbacks. An OAM Cell Loopback is performed (by the ATM NE) by inserting a Loopback OAM cell in the Egress direction, with the Loopback Location field set as specified by the management application, into the cell stream of the VP/VC connection or connection segment, and verifying its return. The following information shall be supplied with each management application request to perform an OAM Cell Loopback test:

- *The Loopback OAM Cell Source Point*

  This is the identity of the ATM VP/VC connection or trail termination point responsible for inserting the Loopback cell.

- *The OAM Cell Loopback Point*

  This is either an indication that the OAM cell loopback is to be performed at the end of the segment/connection, or a code that uniquely identifies an intermediate point along the connection/segment responsible for logically looping-back the received OAM cell.

- *Segment and/or End-to-End Indication*

  This indicates whether the Loopback cell to use during the test shall be a segment OAM cell and/or an end-to-end OAM cell.

The operations interface shall support ATM NE reply messages used to autonomously report the results of a previously requested OAM Cell Loopback test. These results shall be provided to the management application as a Boolean Pass/Fail indication.

6.3.2 Performance information collection

6.3.2.1 VP/VC UPC/NPC information collection

UPC and NPC algorithms are intended to enforce the traffic contract of the incoming cells to ensure that each access connection supported by the ATM NE is complying with prenegotiated traffic descriptors. Since cells discarded due to UPC/NPC functions and cells discarded due to transmission errors and malfunctions will have the same effect on the

end-to-end performance of a VPC/VCC, it is important for trouble shooting and trouble sectionalization purposes to provide network managers with the tools needed to distinguish between these two events.

The following operations interface capabilities are required so that management applications can retrieve ATM NE collected information that reflects the extent to which individual connections are violating their prenegotiated traffic descriptors.

The operations interface shall provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information from selected VP/VC links for which UPC/NPC Disagreement Monitoring is being performed:

- *Discarded Cells due to UPC/NPC Disagreements* (whole stream)

  This parameter provides a count of the number of ATM cells discarded due to traffic descriptor violations detected by the combined CLP=0 and CLP=1 UPC/NPC policing function.

- *Discarded CLP=0 Cells due to UPC/NPC Disagreements* (whole stream)

  This parameter provides a count of the number of high priority (CLP=0) ATM cells discarded due to traffic descriptor violations detected by the CLP=0 UPC/NPC policing function. This counter is only required if CLP=0 traffic is separately policed.

- *Successfully Passed Cells* (whole stream)

  This parameter provides a count of the number of cells that have been passed (i.e. not discarded) by the combined CLP=0 and CLP=1 UPC/NPC policing function.

- *Successfully Passed CLP=0 Cells* (whole stream)

  This parameter provides a count of the number of high priority cells that have been passed (i.e. not discarded) by the CLP=0 UPC/NPC policing function. This counter is only required if CLP=0 traffic is separately policed.

- *Tagged CLP=0 cells* (whole stream)

  This parameter provides a count of the cells which have been tagged.

In addition, as an option, the operations interface should provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information on oam cells UPC/NPC from selected VP/VC links for which UPC/NPC Disagreement Monitoring is being performed:

- *Discarded Cells due to UPC/NPC Disagreements (oam cells only)*

  This parameter provides a count of the number of ATM cells discarded due to traffic descriptor violations detected by the combined CLP=0 and CLP=1 UPC/NPC policing function.

- *Successfully Passed Cells (oam cells only)*

  This parameter provides a count of the number of cells that have been passed (i.e. not discarded) by the combined CLP=0 and CLP=1 UPC/NPC policing function.

6.3.2.2 VP/VC Link Configuration

The operations interface shall provide the management application the ability to initiate VP/VC performance monitoring on a limited number of VP/VC termination points. The data are used by the OS to compute the loss cell ratio and the misinserted cell ratio.

The operations interface shall provide management applications the ability to retrieve current (15-minute/24-hour) counts of the following information from each VP/VC termination for which Performance Monitoring is being performed:

- User Cells.
- Lost Cells.
- Misinserted Cells.

6.3.2.3 VP/VC traffic load information collection

The operations interface shall provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information for selected VP/VC links:

- Ingress cells (whole stream).
- Egress cells (whole stream).

In addition, as an option, the operations interface should provide to the OS the ability to retrieve current counts (15-minute/24-hour) of the following information for selected VP/VC links:

- Ingress cells (oam cells only).
- Egress cells (oam cells only).

6.3.3 Configuration Management

To set up an end-to-end ATM connection, the management application has to configure both VP/VC connection termination points at the input and output ports of the ATM NE and to connect the two connection termination points (cross-connection set-up and release). To do so, the management application may use either a single or multiple requests. Both options are supported by the information model. This subclause uses the vocabulary provided in Recommendation G.805 (connection point, connection, trail termination).

6.3.3.1 Connection Termination Points Configuration

The operations interface shall support the OS requests for the following VP/VC connection termination point configuration capabilities:

– **Create/Delete VP/VC connection termination points**: This capability deals with the creation/deletion of the VP/VC connection termination points at the input and output ports of the ATM NE. The following capabilities are required:

  - identify the VP/VC connection point location;
  - create the VP/VC connection termination points, provide their traffic descriptor and QOS, and activate UPC/NPC based on the traffic descriptors;
  - delete the connection termination points, and deactivate UPC/NPC based on the traffic descriptors and free the corresponding resources (e.g. bandwidth).

– **Modify VP/VC parameters**: This capability deals with the modification of the parameters (such as bandwidth or QOS) of the VP/VC connection termination point at any time. The following capabilities are required:

  - identify the VP/VC connection termination point location;
  - retrieve the VP/VC parameters;
  - modify the VP/VC end-points traffic descriptors or their QOS.

To create or modify a VP/VC connection termination point, the management application shall supply the ATM NE with the following information:

- Ingress and Egress Peak Cell Rate for CLP=0 and CLP=0+1 Traffic (optional).
- Ingress and Egress Sustainable Cell Rate for CLP=0 and CLP=0+1 Traffic (optional).
- Ingress and Egress Burst Tolerance for CLP=0 and CLP=0+1 Traffic (optional).
- Ingress and Egress CDV Tolerance (PCR) for CLP=0 and CLP=0+1 Traffic (optional).
- Ingress and Egress CDV Tolerance (SCR) for CLP=0 and CLP=0+1 Traffic (optional).
- Ingress and Egress QOS class (optional).

The ATM NE management interface capabilities shall enable the OS to configure VPL or VCL end-points as either a segment or non-segment end-point.

The following operations interface capabilities are required to support the configuration and reconfiguration of VPC and VCC segments:

– The operations interface shall support management application requests to configure and reconfigure active VPL and VCL connection termination points as either segment or non-segment end-points.

– The operations interface shall support management application requests to retrieve the information stored in the ATM NE that identifies whether a particular VPL or VCL connection termination point has or has not been configured as a segment end-point.

6.3.3.2 VP/VC Connection Configuration

The operations interface shall support the OS requests for the following VP/VC point-to-point connection configuration capabilities:

– **Set-up and release of VP/VC connection termination**: This capability deals with establishment and release of VP/VC connection termination, and the binding of the input/output of such a connection to VP/VC end-points.

  The following capabilities are required:

  - identify (and possibly create) the VP/VC connection termination point;
  - create the VP/VC connection;
  - bind the VP/VC connection to the VP/VC connection points;
  - set the relevant VP/VC connection parameters;
  - release the VP/VC connection and free the corresponding resources.

  The OS request shall supply:

  - The identification of the VP/VC connection termination points, including or not the vpi/vci value the selection of which may be left to the ATM NE if VP/VC end-points must be created.
  - The relevant VP/VC connection parameters.

– **Modification of VP/VC connection termination**: This capability deals with the modification of VP/VC connection parameters.

  To set up a VP/VC connection, the management application shall supply the ATM NE with the end-points to cross-connect specified as a) or c) for VP connection, and b), c), or d) for VC connection:

  a) The VPI value of a VP connection termination point within a specific ATM Interface.

  b) The VCI value of a VC connection termination point within a specific VPC.

  c) The identity of the supporting ATM Interface termination point (Agent selects the VPI, and if needed VCI value within the ATM Interface).

  d) The identity of the supporting VP connection point (Agent selects the VCI value within the VPC).

– **Administrative state of the VP/VC connection**: The VP/VC connection must have the capability of setting a VP/VC connection in an "unlocked" administrative state (traffic flow is enabled), or in a "locked" state (traffic flow is suspended). This capability will be used by management application to take corrective actions in response to performance degradation or a fault in the cross connection. The administrative state is used to reserve a connection.

6.3.3.3 VP/VC Trail Termination Configuration

The operations interface shall support the OS requests for the following VP/VC trail termination configuration capabilities:

– **Create/delete the VP/VC trail termination point**: This capability deals with the establishment and the release of VP/VP trail termination. The following capabilities are required:

  - create the VP/VC trail termination point;
  - bind the VP/VC trail termination point to a VP/VC connection termination point;
  - set up the relevant trail termination parameters;
  - delete the VP/VC trail termination point;
  - delete the VP/VC connection.

– **Modify the VP/VC trail termination parameters**: This capability deals with the modification of VP/VP trail termination parameters.

7 Management information model

7.1 Information model overview

This Recommendation defines an information model that provides a formal representation of the information exchanged across the CMIP-based interface used to manage ATM Network Elements (NEs).

Figure 7-1 (Parts 1 and 2) illustrates the inheritance relationships between these managed object classes.

Figure 7-2 illustrates the containment relationships between the managed object classes identified here for support of ATM NE management. Note that an arrow pointing from one object class to another object class represents the subordinate/superior relationship used as the basis for naming managed object classes.

Figure 7-3 illustrates the Entity-Relationship Diagram for depicting relationships of the ATM NE management information model.

7.2 Managed Object Class Definitions

7.2.1 atmAccessProfile

atmAccessProfile MANAGED OBJECT CLASS
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
CHARACTERIZED BY "ITU-T M.3100":attributeValueChangeNotificationPackage,
"ITU-T M.3100":createDeleteNotificationsPackage,
atmAccessProfilePkg PACKAGE
BEHAVIOUR atmAccessProfileBeh;
ATTRIBUTES
atmAccessProfileId
GET;;;
CONDITIONAL PACKAGES
vpLevelProfilePackage
PRESENT IF "profiling of the VP level at the ATM interface is supported. This package shall not be present in instances of the atmAccessProfile object that are contained in vpTTPBidirectional.",
vcLevelProfilePackage
PRESENT IF "profiling of the VC level at the ATM interface is supported.",
maxBandwidthPackage
PRESENT IF "the managing system requests it and the managed system
supports it. This package shall only be present when the vpLevelProfilePackage is present.";
REGISTERED AS { i751ObjectClass 1 };

atmAccessProfileBeh BEHAVIOUR
DEFINED AS
"The atmAccessProfile object class is a class of managed objects that characterize the client/server relationship at the VP and/or VC level.

Instances of this managed object class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively. Instances of the atmAccessProfile object class can only be contained in a tcAdaptorTTPBidirectional object to which an instance of the UNI, interNNI, or intraNNI object class points.

Note that if VC level profile information is contained in both the tcAdaptorTTPBidirectional object and the supported vpTTPBidirectional object, then the VC level profiling information in the vpTTPBidirectional object takes precedence over that in the tcAdaptorTTPBidirectional for the VP trail being terminated, if the profiling information in the vpTTPBidirectional object includes attribute values that are lower than those in the tcAdaptorTTPBidirectional . If not, the tcAdaptorTTPBidirectional profiling information applies.

vpTTPBidirectional objects that have not been configured with their own VC level profiling information, would make use of the maxNumVCIBitsNEarEnd and the maxNumVCIBitsSupported information configured for the supporting tcAdaptorTTPBidirectional object.";

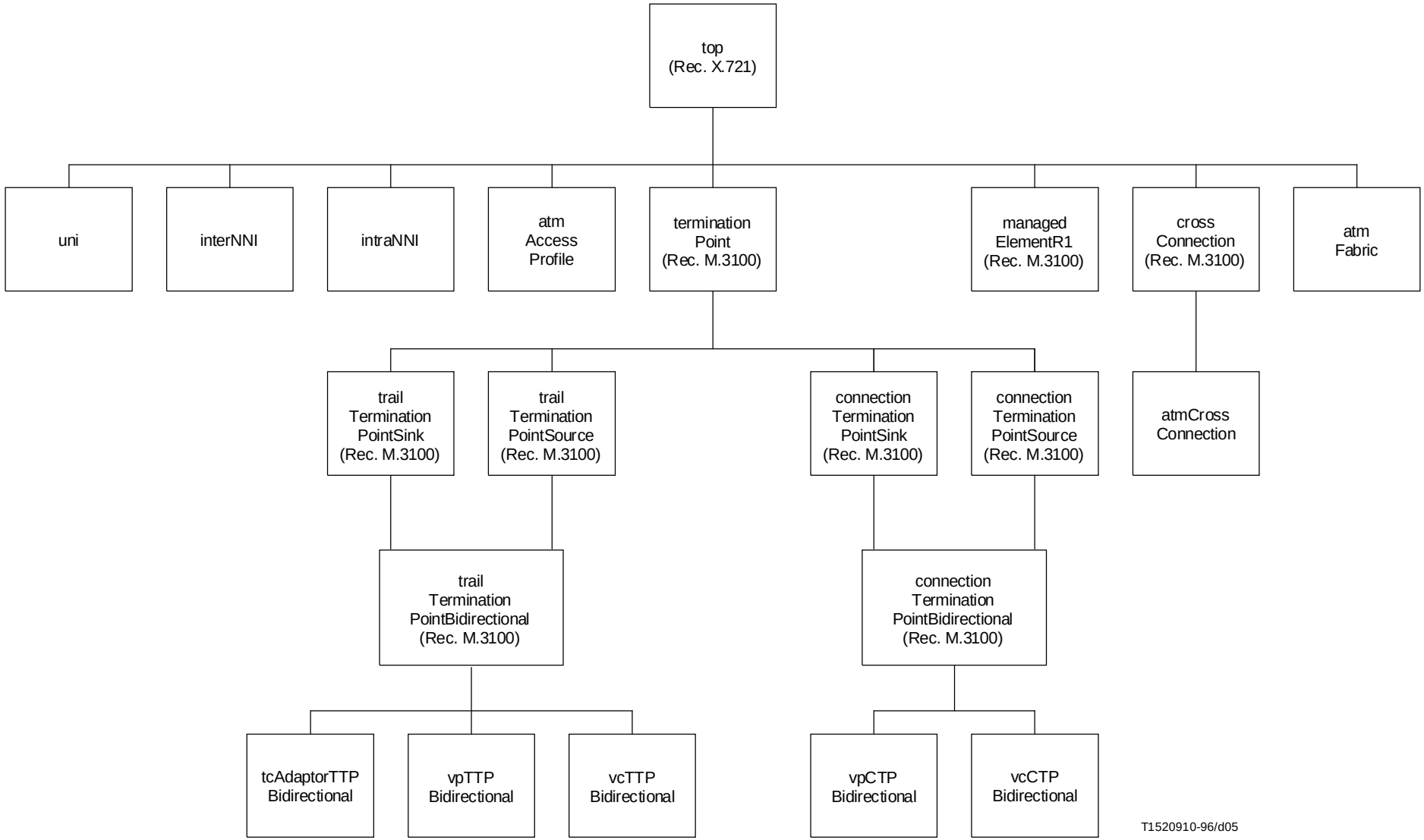


FIGURE 7-1/I.751 *(part 1)*

Inheritance tree for ATM objects

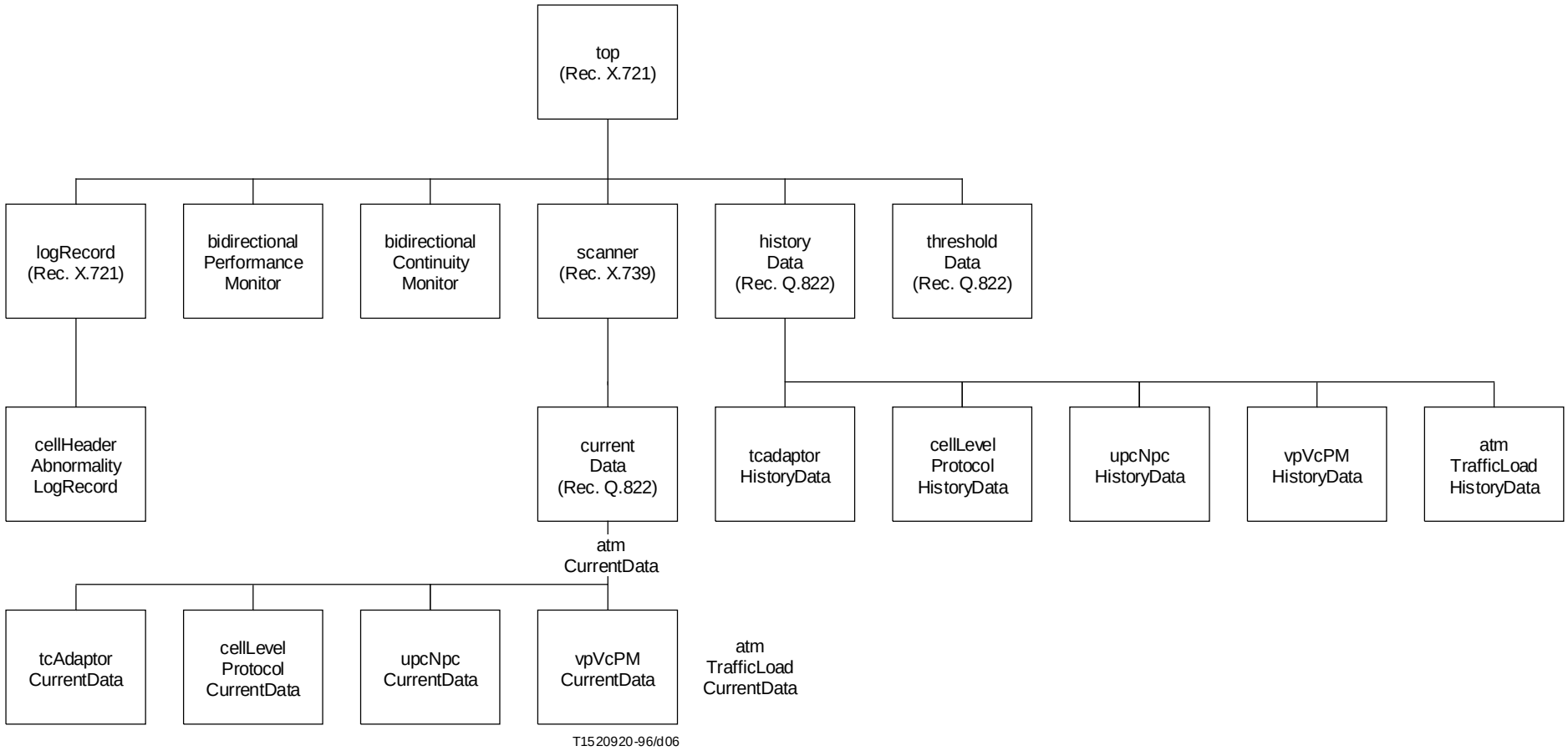


FIGURE 7-1/I.751 *(part 2)*

Inheritance tree for ATM objects

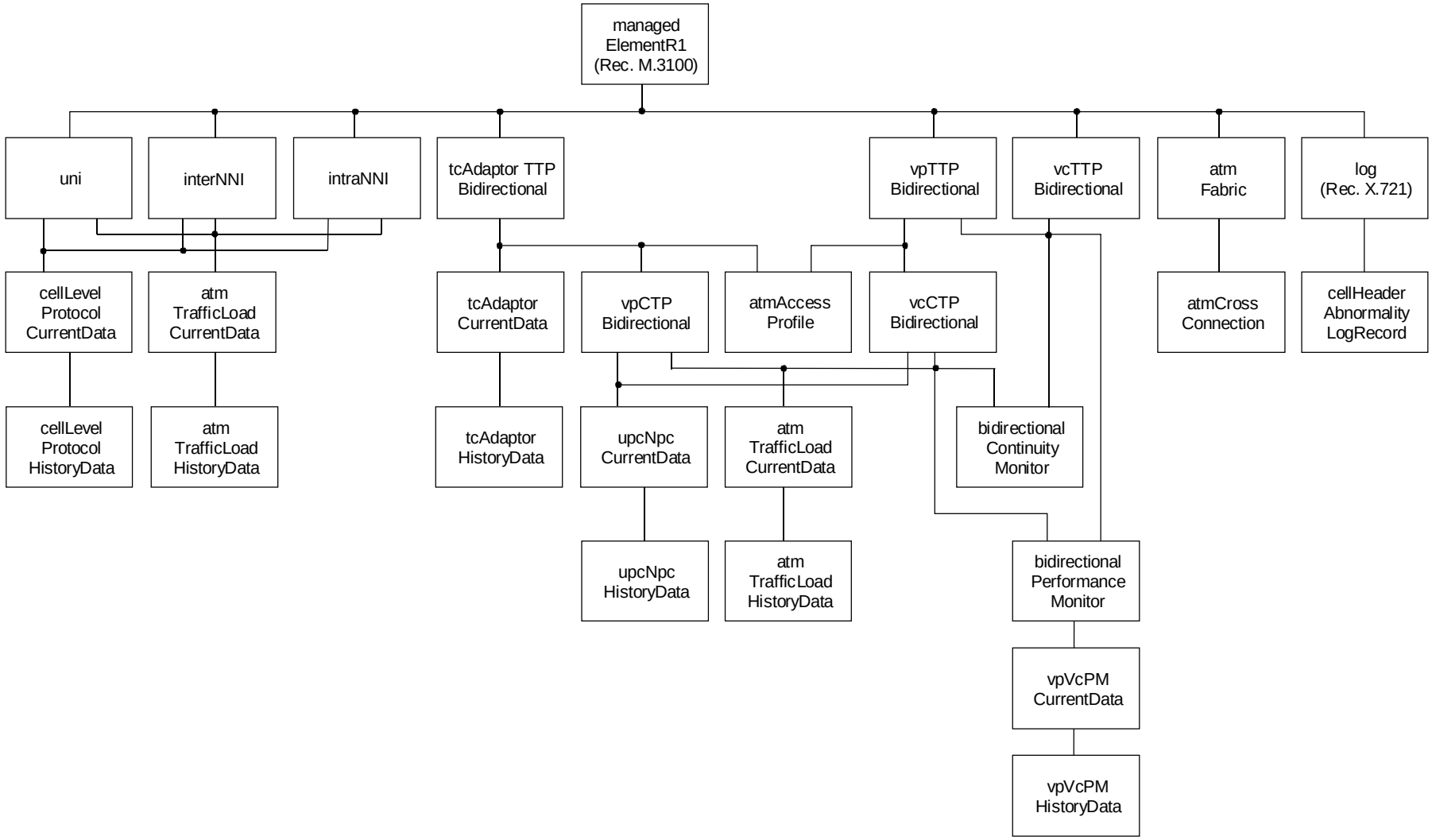


FIGURE 7-2/I.751

Containment tree for ATM objects

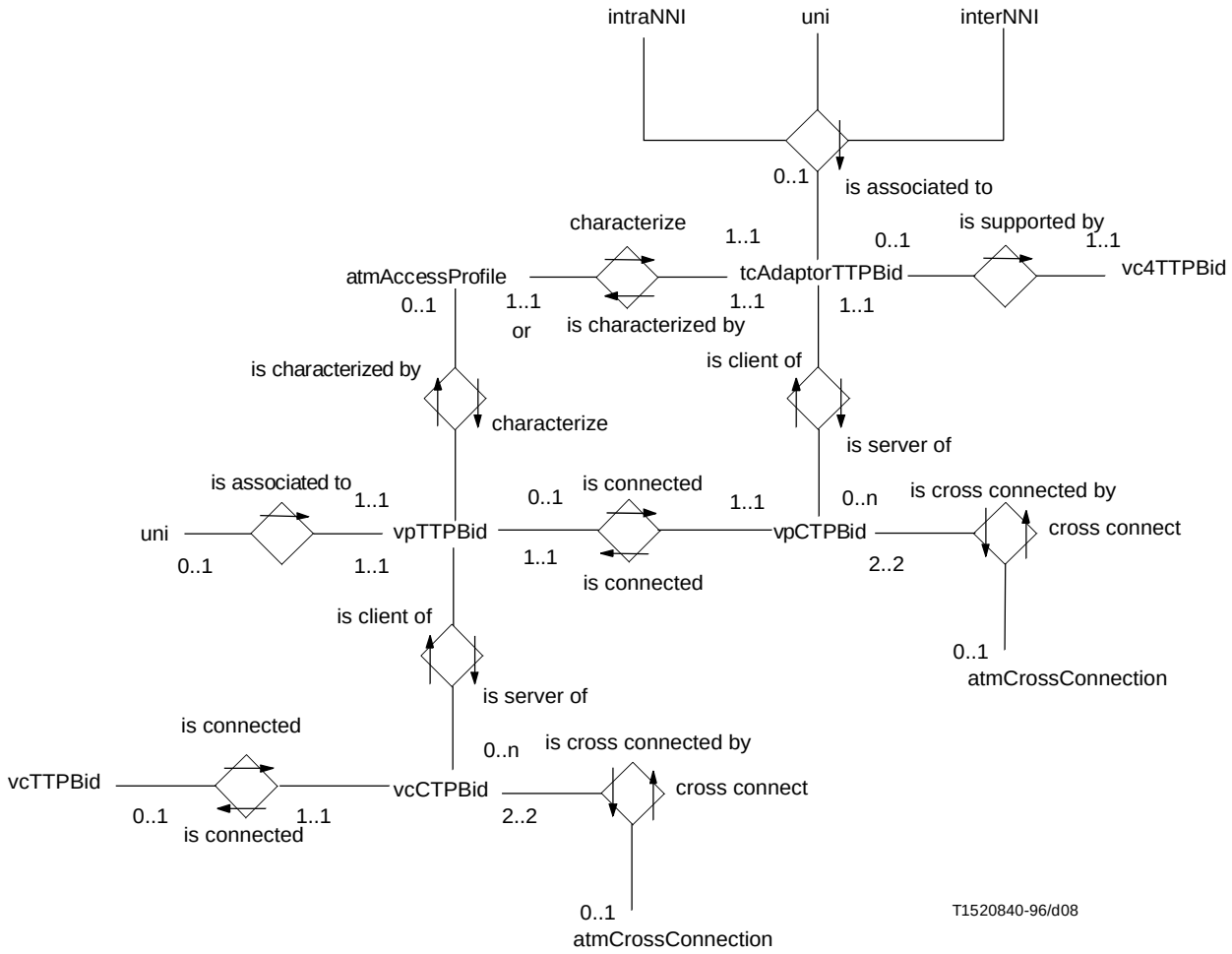


FIGURE 7-3/I.751

ATM Entity-Relationship diagram

7.2.2 atmCrossConnection

atmCrossConnection MANAGED OBJECT CLASS
DERIVED FROM "ITU-T M.3100":crossConnection;
CHARACTERIZED BY
"ITU-T M.3100":createDeleteNotificationsPackage,
atmCrossConnectionPkg PACKAGE
BEHAVIOUR atmCrossConnectionBeh;;;
REGISTERED AS { i751ObjectClass 2 };

atmCrossConnectionBeh BEHAVIOUR
DEFINED AS
"The atmCrossConnection object represents an ATM cross-connect relationship between VP or VC links terminating on the managed system.

For point-to-point ATM VP and VC cross-connections, this managed object identifies the cross-connection relationship between two instances of the vpCTPBidirectional object class or vcCTPBidirectional object class, respectively. The vpCTPBidirectional or vcCTPBidirectional object instances being cross-connected are identified by the fromTermination and toTermination attributes inherited from the crossConnection object class defined in ITU-T Recommendation M.3100.

The administrativeState attribute inherited by this managed object may be used by the managing system to inhibit (lock) and allow (unlock) all ATM cell flow through the ATM cross-connection being represented (i.e. OAM cells and user-data cells).

Instances of this object class are automatically created and deleted by the managed system based on operations performed on the containing atmFabric object.

All ATM VP/VC cross-connections are, by definition, bi-directional; therefore, the directionality attribute, inherited from the crossConnection object class shall be set to the fixed value of bidirectional.

When the value of the administrativeState attribute is LOCKED, the upStreamConnectivityPointer and downStreamConnectivityPointer attributes in the connected CTPs shall be set to NULL, as defined in ITU-T Recommendation M.3100. When the value of the administrativeState attribute is UNLOCKED, the upStreamConnectivityPointer and downstreamConnectivityPointer attributes in the connected CTPs shall be set such that each CTP points to the CTP to which it is connected.

The value of the signalType attribute shall be set by the managed system to vpCI (VP Characteristic Information), when VP cross-connections are being performed, or to vcCI (VC Characteristic Information), when VC cross-connections are being performed.";

7.2.3 atmCurrentData

```
atmCurrentData MANAGED OBJECT CLASS
    DERIVED FROM "Rec. Q.822: 1993":currentData;
    CHARACTERIZED BY
        "Rec. Q.822: 1993":zeroSuppressionPkg,
        atmCurrentDataPkg PACKAGE

            BEHAVIOUR atmCurrentDataBeh;;;
    CONDITIONAL PACKAGES
        "Rec. M.3100": currentProblemListPackage
        PRESENT IF "thresholdPkg is present";
REGISTERED AS { i751ObjectClass 3 };

atmCurrentDataBeh BEHAVIOUR
    DEFINED AS
    "The atmCurrentData object class is used to define generic characteristic for ATM performance monitoring from
        which subclasses are defined in order to hold performance event counts for a specific monitoring point.";
```

7.2.4 atmFabric

```
atmFabric MANAGED OBJECT CLASS
    DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
    CHARACTERIZED BY
        "ITU-T M.3100":stateChangeNotificationPackage,
        atmFabricPackage PACKAGE
            BEHAVIOUR atmFabricBeh;
            ATTRIBUTES
                atmFabricId
                    GET,
                "Rec. X.721 | ISO/IEC-10165-2":administrativeState
                    GET-REPLACE,
                "Rec. X.721 | ISO/IEC-10165-2":operationalState
                    GET,
                "Rec. X.721 | ISO/IEC-10165-2":availabilityStatus
                    GET;
            ACTIONS
                connect,
                disconnect;;;
REGISTERED AS { i751ObjectClass 4 };

atmFabricBeh BEHAVIOUR
    DEFINED AS
    "This object class represents the function of managing the establishment and release of ATM cross-connections.
```

Administrative State:

- **Unlocked: The atmFabric is allowed to perform its normal functions. ACTIONS will be accepted to set up, rearrange or remove cross-connections.**
- **Locked: The atmFabric is not allowed to perform its normal functions. No ACTIONS will be accepted. No new cross-connections can be set up or removed.**

Operational State:

- **Enabled: When the atmFabric is in the enabled operational state, it may be fully operational or partially operational (partially operational is indicated by the availability status attribute).**
- **Disabled: The atmFabric is incapable of performing its normal function. For instance, the managing system will not be able to set up or remove any cross-connections.**

Availability Status:

The supported values for this attribute are:

- **Degraded: The atmFabric is degraded in some respect. For instance, the atmFabric cannot perform the function of establishing new cross-connections while it can still accept ACTIONS to rearrange existing connections. The atmFabric remains available for service (i.e. its operational state is enabled) while it is degraded.**
- **Empty SET (none of the availableStatus conditions exist).**

One instance of the atmFabric object class shall be automatically created by the managed system upon completion of system initialization.";

7.2.5 atmTrafficLoadCurrentData

```
atmTrafficLoadCurrentData MANAGED OBJECT CLASS
    DERIVED FROM atmCurrentData;
    CHARACTERIZED BY
        atmTrafficLoadCurrentDataPackage PACKAGE
            BEHAVIOUR atmTrafficLoadCurrentDataBeh;
            ATTRIBUTES
                incomingCells
                    REPLACE-WITH-DEFAULT
                    DEFAULT VALUE AtmMIBMod.integerZero
                    GET,
                outgoingCells
                    REPLACE-WITH-DEFAULT
                    DEFAULT VALUE AtmMIBMod.integerZero
                    GET;;;
        CONDITIONAL PACKAGES
            oamCellCountingPkg
                PRESENT IF "the managing system requested OAM cell counting
                and the managed system supports it.";

REGISTERED AS { i751ObjectClass 5 };

atmTrafficLoadCurrentDataBeh BEHAVIOUR
    DEFINED AS
    "This managed object contains the current data collected for its superior object. The measurements stored by this
        object are the following:

        – number of incoming cells
        – number of outgoing cells
        – number of incoming OAM cells (conditionally present)
        – number of outgoing OAM cells (conditionally present)

    Instances of this object class are created by the managing system using the CMIS M-CREATE service. Instances of this
        object class may also be deleted either using the CMIS M-DELETE service, or automatically by the managed
        system when the containing object instance is deleted.";
```

7.2.6 atmTrafficLoadHistoryData

```
atmTrafficLoadHistoryData MANAGED OBJECT CLASS
    DERIVED FROM "Rec. Q.822:1993":historyData;
    CHARACTERIZED BY
        atmTrafficLoadHistoryDataPackage PACKAGE
            BEHAVIOUR atmTrafficLoadHistoryDataBeh;
            ATTRIBUTES
                incomingCells
                    GET,
                outgoingCells
                    GET;;;
```

CONDITIONAL PACKAGES

incomingOamCellsHistoryDataPkg

PRESENT IF "supplied by atmTrafficLoadCurrentData",

outgoingOamCellsHistoryDataPkg

PRESENT IF "supplied by atmTrafficLoadCurrentData";

REGISTERED AS { i751ObjectClass 6 };

atmTrafficLoadHistoryDataBeh BEHAVIOUR

DEFINED AS

"Instances of the atmTrafficLoadHistoryData object class are used to store the observed events of a atmTrafficLoadCurrentData object at the end of the 15-minute/24-hour observation interval. Instances of this managed object class are contained by an instance atmTrafficLoadCurrentData managed object class.";

7.2.7 bidirectionalContinuityMonitor

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.";

7.2.8 bidirectionalPerformanceMonitor

```
bidirectionalPerformanceMonitor MANAGED OBJECT CLASS
DERIVED FROM "Rec. X.721 | ISO/IEC-10165-2":top;
CHARACTERIZED BY
    "ITU-T M.3100":stateChangeNotificationPackage,
    "ITU-T M.3100":attributeValueChangeNotificationPackage,
    bidirectionalPerformanceMonitorPackage PACKAGE
        BEHAVIOUR bidirectionalPerformanceMonitorBeh;
        ATTRIBUTES
            performanceMonitorId
                GET,
            sinkPMMechanismActive
                GET,
            sourcePMMechanismActive
                GET,
            inAverageMonitoringBlockSize
                GET,
            outAverageMonitoringBlockSize
                GET,
            "Rec. X.721 | ISO/IEC-10165-2":operationalState
                GET;
            ACTIONS
                controlPM;;;
        CONDITIONAL PACKAGES
            flowDirectionPkg
                PRESENT IF "the monitor object instance is contained in CTP",
            monitoredFlowPkg
                PRESENT IF "non-intrusive performance monitoring is supported",
            backwardPMReportingPkg
                PRESENT IF "an instance supports it",
            fePMDataCollectionPkg
                PRESENT IF "an instance supports it";
REGISTERED AS { i751ObjectClass 8 };

bidirectionalPerformanceMonitorBeh BEHAVIOUR
    DEFINED AS
    "This object is instanciated for non-intrusive PM and for intrusive PM. In the case of intrusive PM the F4/F5 OAM
        flow is extracted, while in the case of non-intrusive PM the F4/F5 flow is only observed. If the
        monitoredFlowPkg is supported in the M-Create request the bidirectionalPerformanceMonitor will be non-
        intrusive. If the monitoredFlowPkg is not supported in the M-Create request the
        bidirectionalPerformanceMonitor will be intrusive.

    When this object is instanciated for non-intrusive PM, only the sink mechanism will be activated. The
        monitoredFlowPkg is mandatory for non-intrusive PM and can be either segment or end-to-end, according to
        the value of the monitoredFlow attribute. At the CTP, two non-intrusive bidirectionalPerformanceMonitor
        objects should be created if the two directions of a bidirectional VP or VC must be monitored. The controlPM
        action is used to also request both the activation and the deactivation of both the non-intrusive PM end-to-end
        and segment OAM cell processing procedures at the CTP.

    When the managing system creates the bidirectionalPerformanceMonitoring object for intrusive monitoring, it shall
        supply the monitoring block size. When the managing system creates the bidirectionalPerformanceMonitoring
        object for non-intrusive monitoring, it shall not supply the monitoring block size.

    The non-intrusive performance monitoring function can apply to forward monitoring OAM cells and/or backward
        reporting OAM cells.

    This managed object models a performance monitoring OAM flow. For intrusive PM, when this object is named by a
        vpCTPBidirectional or vcCTPBidirectional object, then it represents the termination of a segment OAM flow.
```

The controlPM action may be used to request the object to start a performance monitoring procedure (intrusive and non-intrusive PM). The controlPM action allows to separately activate the sink and source mechanisms (in the bidirectionalPerformanceMonitor object) which will be reflected by setting to TRUE the values of the sinkPMMechanismActive and sourcePMMechanismActive attributes, respectively.

If far-end data collection is supported, the controlPM action may also be used to request the collection of the far-end PM data received from the remote segment termination point. If the request is accepted, the fePMDataCollectionMechanismActive attribute will be set to TRUE.

If backward data reporting is supported, the controlPM action may also be used to activate the backward reporting PM OAM cell mechanism. When the request is accepted, the backwardPMReportingMechanismActive attribute will be set to TRUE.

The managing system can request the creation of the bidirectionalPerformanceMonitor object instance and the activation of the Performance Monitoring mechanism at the same time in the CREATE request by setting the sinkPMMechanismActive attribute (and possibly the backwardPMReportingMechanismActive attribute) to TRUE.

When the managing system requests the creation of a bidirectionalPerformanceMonitor object instance without the activation of Performance Monitoring, the managed system shall set the sinkPMMechanismActive attribute (and the backwardPMReportingMechanismActive attribute, if applicable) to FALSE.

The managing system can request the creation of the bidirectionalPerformanceMonitor object instance and the activation of the Performance Monitoring mechanism at the same time in the CREATE request by setting the sourcePMMechanismActive attribute (and possibly the fePMDataProcessingMechanismActive attribute) to TRUE.

When the managing system requests the creation of a bidirectionalPerformanceMonitor object instance without the activation of Performance Monitoring, the managed system shall set the sourcePMMechanismActive attribute (and the fePMMechanismActive attribute, if applicable) to FALSE.

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.";

7.2.9 cellHeaderAbnormalityLogRecord

```
cellHeaderAbnormalityLogRecord MANAGED OBJECT CLASS
    DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":logRecord;
    CHARACTERIZED BY
        cellHeaderAbnormalityLogRecordPkg PACKAGE
            BEHAVIOUR cellHeaderAbnormalityLogRecordBeh;
            ATTRIBUTES
                cellHeaderAbnormalityType
                    GET,
                interfacePointer
                    GET,
                vpiValue
                    GET,
                vciValue
                    GET;;;
REGISTERED AS { i751ObjectClass 9 };

cellHeaderAbnormalityLogRecordBeh BEHAVIOUR
    DEFINED AS
    "The cellHeaderAbnormalityLogRecord object class is a class of managed support objects used to log information that
        describes ATM cell header protocol abnormality events detected by the managed system.

    Attributes have been defined so that each record of the log conveys the following information:

        – Abnormality Type
          (not assigned VPI/VCI Value or Out-of-Range VPI/VCI Value)
        – VPI/VCI Value
        – ATM Interface ID
          (i.e. pointer to the associated uni, interNNI, or intraNNI object)
        – Date and Time of Log Entry.";
```

7.2.10 cellLevelProtocolCurrentData

```
cellLevelProtocolCurrentData MANAGED OBJECT CLASS
        DERIVED FROM atmCurrentData;
        CHARACTERIZED BY
            "ITU-T M.3100":
            createDeleteNotificationsPackage,
            cellLevelProtocolCurrentDataPkg PACKAGE
                BEHAVIOUR cellLevelProtocolCurrentDataBeh;
                ATTRIBUTES
                    discardedCellsInvalidHeader
                        REPLACE-WITH-DEFAULT
                        DEFAULT VALUE AtmMIBMod.integerZero
                        GET;;;
REGISTERED AS { i751ObjectClass 10 };

cellLevelProtocolCurrentDataBeh BEHAVIOUR
        DEFINED AS
        "Instances of the cellLevelProtocolCurrentData object class are used to hold the current (15-minute/24-hour) register
                counts reflecting the protocol monitoring functions performed per ATM UNI, Inter-NNI, and Intra-NNI.

        Each instance of this managed object shall maintain a thresholded count of the number of cells discarded due to the
                detection of ATM Layer protocol violations (invalid header pattern, not assigned VPI/VCI value, or out-of-
                range VPI/VCI value).

        Instances of this object class shall be inherently created by the managed system whenever an instance of the uni,
                interNNI, or intraNNI object class is created.

        This managed object class uses the cellLevelProtocolHistoryData managed object class for history retention.";
```

7.2.11 cellLevelProtocolHistoryData

```
cellLevelProtocolHistoryData MANAGED OBJECT CLASS
        DERIVED FROM "Rec. Q.822: 1993":historyData;
        CHARACTERIZED BY
            cellLevelProtocolHistoryDataPkg PACKAGE
                BEHAVIOUR cellLevelProtocolHistoryDataBeh;
                ATTRIBUTES
                    discardedCellsInvalidHeader
                        GET;;;
REGISTERED AS { i751ObjectClass 11 };

cellLevelProtocolHistoryDataBeh BEHAVIOUR
        DEFINED AS
        "Instances of the cellLevelProtocolHistoryData object class are used to store the observed events of a
                cellLevelProtocolCurrentData object at the end of the 15-minute/24-hour observation interval. Instances of this
                managed object class are contained by an instance cellLevelProtocolCurrentData managed object class.";
```

7.2.12 interNNI

```
interNNI MANAGED OBJECT CLASS
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
        CHARACTERIZED BY
            interNNIPkg PACKAGE
                BEHAVIOUR interNNIBeh;
                ATTRIBUTES
                    interNNIId
                        GET,
                    underlyingTTPPointer
                        GET;;;
            CONDITIONAL PACKAGES
                "ITU-T M.3100":createDeleteNotificationsPackage
                    PRESENT IF "the objectCreation and objectDeletion notifications defined in
                    Recommendation X.721 are supported by an instance of this class.",
                loopbackLocationIdentifierPkg
                    PRESENT IF "supplied by the managing system";
REGISTERED AS { i751ObjectClass 12 };
```

interNNIBeh BEHAVIOUR
DEFINED AS
"This managed object is used to configure and identify an ATM interface on the managed system as an Inter-NNI.

The underlyingTTPPointer attribute provides a pointer relationship to the tcAdaptorTTPBidirectional object that represents the location in the managed system where ATM is adapted to the physical transmission path.

An instance of this object class shall exist for each Inter-NNI terminating on the managed system. Instances of this object class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively.

The loopbackLocationIdentifierPkg provides a read/write code used for OAM cell loopback purposes. Incoming OAM Loopback cells with a Loopback Location field value that matches the value of the loopbackLocationIdentifier attribute shall be looped-back over the Inter-NNI.";

7.2.13 intraNNI

intraNNI MANAGED OBJECT CLASS
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
CHARACTERIZED BY
intraNNIPkg PACKAGE
BEHAVIOUR intraNNIBeh;
ATTRIBUTES
intraNNIId
GET,
underlyingTTPPointer
GET;;;
CONDITIONAL PACKAGES
"ITU-T M.3100":createDeleteNotificationsPackage
PRESENT IF "the objectCreation and objectDeletion notifications defined in Recommendation X.721 are supported by an instance of this class.",
loopbackLocationIdentifierPkg
PRESENT IF "supplied by the managing system";
REGISTERED AS { i751ObjectClass 13 };

intraNNIBeh BEHAVIOUR
DEFINED AS
"This managed object is used to configure and identify an ATM interface on the managed system as a Intra-NNI.

The underlyingTTPPointer attribute provides a pointer relationship to the tcAdaptorTTPBidirectional object that represents the location in the managed system where ATM is adapted to the physical transmission path.

An instance of this object class shall exist for each Intra-NNI terminating on the managed system. Instances of this object class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively.

The loopbackLocationIdentifierPkg provides a read/write code used for OAM cell loopback purposes. Incoming OAM Loopback cells with a Loopback Location field value that matches the value of the loopbackLocationIdentifier attribute shall be looped-back over the Intra-NNI.";

7.2.14 tcAdaptorCurrentData

tcAdaptorCurrentData MANAGED OBJECT CLASS
DERIVED FROM atmCurrentData;
CHARACTERIZED BY
"ITU-T M.3100":createDeleteNotificationsPackage,
tcAdaptorCurrentDataPkg PACKAGE
BEHAVIOUR tcAdaptorCurrentDataBeh;
ATTRIBUTES
discardedCellsHECViolation
REPLACE-WITH-DEFAULT
DEFAULT VALUE AtmMIBMod.integerZero
GET,
erroredCellsHECViolation
REPLACE-WITH-DEFAULT
DEFAULT VALUE AtmMIBMod.integerZero
GET;;;
REGISTERED AS { i751ObjectClass 14 };

tcAdaptorCurrentDataBeh BEHAVIOUR

DEFINED AS

"This managed object contains the current protocol monitoring data collected for its superior tcAdaptorTTPBidirectional object. Specifically, this managed object identifies the number of received cells for which an HEC error was detected as well as the number of received cells that were discarded due to HEC errors during the current 15-minute/24-hour observation interval.

This object shall be automatically created whenever an instance of the tcAdaptorTTPBidirectional object class is created.";

7.2.15 tcAdaptorHistoryData

```
tcAdaptorHistoryData MANAGED OBJECT CLASS
    DERIVED FROM "Rec. Q.822: 1993":historyData;
    CHARACTERIZED BY
        tcAdaptorHistoryDataPkg PACKAGE
            BEHAVIOUR tcAdaptorHistoryDataBeh;
            ATTRIBUTES
                discardedCellsHECViolation
                    GET,
                erroredCellsHECViolation
                    GET;;;
REGISTERED AS { i751ObjectClass 15 };

tcAdaptorHistoryDataBeh BEHAVIOUR
    DEFINED AS
    "Instances of the tcAdaptorHistoryData object class are used to store the observed events of a tcAdaptorCurrentData
        object at the end of the 15-minute/24-hour observation interval. Instances of this managed object class are
        contained by an instance of the tcAdaptorCurrentData object class.";
```

7.2.16 tcAdaptorTTPBidirectional

```
tcAdaptorTTPBidirectional MANAGED OBJECT CLASS
    DERIVED FROM "ITU-T M.3100":trailTerminationPointBidirectional;
    CHARACTERIZED BY
        "ITU-T M.3100":tmnCommunicationsAlarmInformationPackage,
        "ITU-T M.3100":createDeleteNotificationsPackage,
        "ITU-T M.3100":stateChangeNotificationPackage,
        tcAdaptorTTPBidirectionalPkg PACKAGE
            BEHAVIOUR tcAdaptorTTPBidirectionalBeh;
            ATTRIBUTES
                tcTTPId
                    GET;;;
        CONDITIONAL PACKAGES
            cellScramblingEnabledPkg
                PRESENT IF "cell scrambling may be activated and deactivated for the
                supporting ATM interface.";
REGISTERED AS { i751ObjectClass 16 };

tcAdaptorTTPBidirectionalBeh BEHAVIOUR
    DEFINED AS
    "This managed object represents a point in the managed system where the adaptation of the ATM Layer to the
        underlying physical infrastructure (e.g. SDH or PDH transport network) takes place. ITU-T Recommendation
        I.321[12] identifies this adaptation function as one of many functions performed at the Transmission
        Convergence (TC) Sublayer of the B-ISDN protocol stack.

    This object is responsible for generating communicationsAlarm notifications that report the inability of the managed
        system to delineate ATM cells from the payload of a terminated digital transmission path.

    The supportedByObjectList attribute inherited from the trailTerminationPoint managed object shall include a pointer
        to the underlying, path-level trail termination point managed object (e.g. vc4TTPBidirectional object).

    Instances of this managed object class shall be automatically created and deleted by the managed system. Instances of
        this object class shall not be deleted when referenced by an instance of the uni, interNNI, or intraNNI object via
        the underlyingTTPPointer attribute. Instances of this object class shall not be deleted unless all contained
        objects are deleted.
```

The tcAdaptorTTPBidirectional object must contain an instance of the atmAccessProfile object, or be associated with an instance of UNI, interNNI, or intraNNI object before it can contain vpCTPBidirectional objects.

A change in the ATM interface type associated with the tcAdaptorTTPBidirectional (i.e. UNI, interNNI, or intraNNI) is prohibited unless the contained atmAccessProfile and all contained vpCTPBidirectional objects are deleted.";

7.2.17 uni

```
uni MANAGED OBJECT CLASS
    DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":top;
    CHARACTERIZED BY
        uniPkg PACKAGE
            BEHAVIOUR uniBeh;
            ATTRIBUTES
                uniId
                    GET,
                underlyingTTPPointer
                    GET;;;
        CONDITIONAL PACKAGES
            "ITU-T M.3100":createDeleteNotificationsPackage
                PRESENT IF "the objectCreation and objectDeletion notifications defined
                in Recommendation X.721 are supported by an instance of this class.",
            loopbackLocationIdentifierPkg
                PRESENT IF "supplied by the managing system";
REGISTERED AS { i751ObjectClass 17 };

uniBeh BEHAVIOUR
    DEFINED AS
    "This managed object is used to configure and identify an ATM interface on the managed system as a User Network
        Interface (UNI).

    The underlyingTTPPointer attribute provides a pointer relationship to the tcAdaptorTTPBidirectional object or
        vpTTPBidirectional object.

    An instance of this object class shall exist for each UNI terminating on the managed system. Instances of this object
        class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services,
        respectively.

    The loopbackLocationIdentifierPkg provides a read/write code used for OAM cell loopback purposes. Incoming OAM
        Loopback cells with a Loopback Location field value that matches the value of the loopbackLocationIdentifier
        attribute shall be looped-back over the UNI.

    If the underlyingTTPPointer attribute points to an instance of the vpTTPBidirectional object class, then the uni object
        shall not contain an instance of the atmTrafficLoadCurrentData object class.";
```

7.2.18 upcNpcCurrentData

```
upcNpcCurrentData MANAGED OBJECT CLASS
    DERIVED FROM atmCurrentData;
    CHARACTERIZED BY
        upcNpcCurrentDataPkg PACKAGE
            BEHAVIOUR upcNpcCurrentDataBeh;
            ATTRIBUTES
                discardedCells
                    REPLACE-WITH-DEFAULT
                    DEFAULT VALUE AtmMIBMod.integerZero
                    GET,
                successfullyPassedCells
                    REPLACE-WITH-DEFAULT
                    DEFAULT VALUE AtmMIBMod.integerZero
                    GET;;;
        CONDITIONAL PACKAGES
            discardedCLP0CellsPkg
                PRESENT IF "the managed system performs UPC/NPC functions
                separately for high Cell Loss Priority (CLP) cells (i.e. cells with
                CLP=0)",
            oamDiscardedCellsPkg
                PRESENT IF "the managing system requests it for OAM cell flow and the
                managed system supports it",
```

```
            oamSuccessfullyPassedCellsPkg
                PRESENT IF " if the managing system requests it for OAM cell flow and
                the managed system supports it",
            successfullyPassedCLP0CellsPkg
                PRESENT IF " if the managed system supports high priority only policing
                and has the ability to count cells that are successfully passed by the CLP=0
                UPC/NPC policing function",
            taggedCLP0CellsPkg
                PRESENT IF "the managed system supports Cell Loss Priority (CLP)
                tagging";
REGISTERED AS { i751ObjectClass 18 };

upcNpcCurrentDataBeh BEHAVIOUR
    DEFINED AS
    "An instance of this managed object class is used to collect current data associated with UPC/NPC functions performed
        on its superior managed object.

    Instances of this object class shall be created by the managing system.

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

    This managed object class uses the upcNpcHistoryData managed object class for history retention.";
```

7.2.19 upcNpcHistoryData

```
upcNpcHistoryData MANAGED OBJECT CLASS
    DERIVED FROM "Rec. Q.822: 1993":historyData;
    CHARACTERIZED BY
        upcNpcHistoryDataPkg PACKAGE
            BEHAVIOUR upcNpcHistoryDataBeh;
            ATTRIBUTES
                discardedCells
                    GET,
                successfullyPassedCells
                    GET;;;
        CONDITIONAL PACKAGES
            discardedCLP0CellsHistoryDataPkg
                PRESENT IF "supplied by upcNpcCurrentData",
            oamDiscardedCellsHistoryDataPkg
                PRESENT IF "supplied by upcNpcCurrentData",
            oamSuccessfullyPassedCellsHistoryDataPkg
                PRESENT IF "supplied by upcNpcCurrentData",
            successfullyPassedCLP0CellsHistoryDataPkg
                PRESENT IF "supplied by upcNpcCurrentData",
            taggedCLP0CellsHistoryDataPkg
                PRESENT IF "supplied by upcNpcCurrentData";
REGISTERED AS { i751ObjectClass 19 };

upcNpcHistoryDataBeh BEHAVIOUR
    DEFINED AS
    "Instances of the upcNpcHistoryData object class are used to store the observed events of an upcNpcCurrentData
        object at the end of the 15-minute/24-hour observation interval. An instance of this managed object is contained
        by a upcNpcCurrentData object instance.";
```

7.2.20 vcCTPBidirectional

```
vcCTPBidirectional MANAGED OBJECT CLASS
    DERIVED FROM "ITU-T M.3100":connectionTerminationPointBidirectional;
    CHARACTERIZED BY
        "ITU-T M.3100":attributeValueChangeNotificationPackage,
        "ITU-T M.3100":createDeleteNotificationsPackage,
        "ITU-T M.3100":crossConnectionPointerPackage,
        vcCTPBidirectionalPkg PACKAGE
            BEHAVIOUR vcCTPBidirectionalBeh;
            ATTRIBUTES
                vcCTPId
                    GET,
                segmentEndPoint
                    DEFAULT VALUE AtmMIBMod.booleanFalseDefault
                    GET-REPLACE;;;
```

CONDITIONAL PACKAGES

egressTrafficDescriptorPkg

PRESENT IF "supplied by the managing system. This package must be present at points where egress UPC/NPC functions are performed.",

ingressTrafficDescriptorPkg

PRESENT IF "supplied by the managing system. This package must be present at points where ingress UPC/NPC functions are performed.",

oamEgressTrafficDescriptorPkg

PRESENT IF "supplied by the managing system. This package must be present at points where egress OAM UPC/NPC functions are performed.",

oamIngressTrafficDescriptorPkg

PRESENT IF "supplied by the managing system. This package must be present at points where ingress OAM UPC/NPC functions are performed.",

"Rec. X.721 | ISO/IEC 10165-2":administrativeStatePackage

PRESENT IF "supported by the managed system",

qosClassesPkg

PRESENT IF "QOS Class information is supplied by the managing system",

loopbackOAMCellPkg

PRESENT IF "the link termination point supports OAM cell Loopbacks";

REGISTERED AS { i751ObjectClass 20 };

vcCTPBidirectionalBeh BEHAVIOUR

DEFINED AS

"The vcCTPBidirectional object class is a class of managed objects that delimit Virtual Channel (VC) links. From a configuration management perspective, instances of this object class represent VC link terminations that are either (1) cross-connected to other VC link terminations, or are available for such cross-connection, or (2) associated to a vcTTPBidirectional instance or are available for such an association.

Instances of this object class include attributes that describe the VCI value, traffic descriptors, and, optionally, the Quality of Service (QOS) class assigned to the VCL termination being represented. Note that the vcCTPId attribute value identifies the VCI value for the VCL being terminated and is also used as the RDN for naming instances of this object class. The vcCTPId attribute value may be provided by the managing system upon creation of this managed object instance or it may be absent in the M-CREATE message and thus selected by the managed system. When selected by the managed system, the value chosen shall be reported to the managing system as a parameter in the response to the successfully carried out M-CREATE request.

From a performance and fault management perspective, instances of this object class represent logical points along VCCs at which various maintenance and network traffic management functions may be performed.

When an instance of this object is configured as a segment end-point (i.e. segmentEndPoint is TRUE), it represents a logical point in the managed system where the segment F5 flow (i.e. OAM cells with PTI=4) for the VC terminates.

In the event that the related vcTTPBidirectional is created, this instance points to the vcTTPBidirectional and its crossConnectionPointer points to the atmFabric instance.

The conditional package loopbackOAMCellPkg provides the M-ACTION used to request the termination point to insert an OAM cell for downstream loopbacking and report whether or not the cell was returned within the required time.

Note that, when configured as a segment end-point, all Segment F5 Loopback cells with a default Loopback Location field value of all ones would be looped-back at this point.

When a VC-AIS or VC-RDI failure is detected, the vcCTPBidirectional object shall generate a communicationsAlarm notification (if the tmnCommunicationsAlarmInformationPackage is present) with the probableCause parameter value set equal to aIS or farEndReceiveFailure, respectively.

The administrativeState attribute may be used by the management system to inhibit (lock) and allow (unlock) the flow of cells through the vcCTPBidirectional. However, when the vcCTPBidirectional object is configured as a segment end-point, the value of the administrativeState attribute shall not affect the ability of the vcCTPBidirectional to perform segment OAM cell processing functions.

Instances of this object class may be created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively. Instances of this managed object class may also be automatically created by the managed system in response to actions performed on instances of the atmFabric object class.";

7.2.21 vcTTPBidirectional

```
vcTTPBidirectional MANAGED OBJECT CLASS
    DERIVED FROM "ITU-T M.3100":trailTerminationPointBidirectional;
    CHARACTERIZED BY
        "Rec. X.721 | ISO/IEC 10165-2":administrativeStatePackage,
        "ITU-T M.3100":attributeValueChangeNotificationPackage,
        "ITU-T M.3100":createDeleteNotificationsPackage,
        vcTTPBidirectionalPkg PACKAGE
            BEHAVIOUR vcTTPBidirectionalBeh;
            ATTRIBUTES
                vcTTPId
                    GET;;;
        CONDITIONAL PACKAGES
            loopbackOAMCellPkg
                PRESENT IF "the VCC termination point supports OAM cell Loopbacks";
REGISTERED AS { i751ObjectClass 21 };

vcTTPBidirectionalBeh BEHAVIOUR
    DEFINED AS
    "The vcTTPBidirectional object class is a class of managed objects that delimit Virtual Channel Connections (VCCs).

    An instance of this object class represents the logical point in the managed system where the end-to-end F5 flow (i.e. OAM cells with PT=5) terminates.

    The conditional package loopbackOAMCellPkg provides the M-ACTION used to request the termination point to insert an OAM cell for downstream loopbacking and report whether or not the cell was returned within the required time.

    Note that all End-to-End F5 Loopback cells with a default Loopback Location field value of all ones would be looped-back at this point.

    Instances of this object class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively.

    An instance of this object shall always point to vcCTPBidirectional managed object using UCP/DCP.";
```

7.2.22 vpCTPBidirectional

```
vpCTPBidirectional MANAGED OBJECT CLASS
    DERIVED FROM "ITU-T M.3100":connectionTerminationPointBidirectional;
    CHARACTERIZED BY
        "ITU-T M.3100":attributeValueChangeNotificationPackage,
        "ITU-T M.3100":createDeleteNotificationsPackage,
        "ITU-T M.3100":crossConnectionPointerPackage,
        vpCTPBidirectionalPkg PACKAGE
            BEHAVIOUR vpCTPBidirectionalBeh;
            ATTRIBUTES
                vpCTPId
                    GET,
                segmentEndPoint
                    DEFAULT VALUE AtmMIBMod.booleanFalseDefault
                    GET-REPLACE;;;
        CONDITIONAL PACKAGES
            egressTrafficDescriptorPkg
                PRESENT IF    "supplied by the managing system. This package must be
                present at points where egress UPC/NPC functions are performed.",
            ingressTrafficDescriptorPkg
                PRESENT IF    "supplied by the managing system. This package must be
                present at points where ingress UPC/NPC functions are performed.",
            oamEgressTrafficDescriptorPkg
                PRESENT IF    "supplied by the managing system. This package must be
                present at points where egress OAM UPC/NPC functions are performed.",
```

oamIngressTrafficDescriptorPkg
PRESENT IF "supplied by the managing system. This package must be present at points where ingress OAM UPC/NPC functions are performed.",
"Rec. X.721 | ISO/IEC 10165-2":administrativeStatePackage
PRESENT IF "supported by the Network Element",
qosClassesPkg
PRESENT IF "QOS Class information is supplied by the managing system",
loopbackOAMCellPkg
PRESENT IF "the VPL termination point supports OAM cell Loopbacks";
REGISTERED AS { i751ObjectClass 22 };

vpCTPBidirectionalBeh BEHAVIOUR
DEFINED AS
"The vpCTPBidirectional object class is a class of managed objects that delimits Virtual Path (VP) links. From a configuration management perspective, instances of this object class represent VP link terminations that are either (1) cross-connected to other VP link terminations, or are available for such cross-connection, or (2) associated to a vpTTPBidirectional instance or are available for such an association.

Instances of this object class include attributes that describe the VPI value, traffic descriptors, and, optionally, the Quality of Service (QOS) class assigned to the VPL termination being represented. Note that the vpCTPId attribute value identifies the VPI value of the VPL termination being represented and is also used as the RDN for naming instances of this object class. Note that the vpCTPId attribute may be provided by the managing system upon creation of this managed object instance or it may be absent in the M-CREATE message and thus selected by the managed system. When selected by the managed system, the value chosen shall be reported to the managing system as a parameter in the response to the successfully carried out M-CREATE request.

From a performance and fault management perspective, instances of this object class represent logical points along VPCs at which various maintenance and network traffic management functions may be performed.

When an instance of this object is configured as a segment end-point (i.e. segmentEndPoint is TRUE), it represents a logical point in the managed system where the segment F4 flow (i.e. OAM cells with VCI=3) for the VP terminates.

In the event that the related vpTTPBidirectional is created, this instance points to the vpTTPBidirectional and its crossConnectionPointer points to the atmFabric instance.

The conditional package loopbackOAMCellPkg provides the M-ACTION used to request the termination point to insert an OAM cell for downstream loopbacking and report whether or not the cell was returned within the required time.

Note that, when configured as a segment end-point, all Segment F4 Loopback cells with a default Loopback Location field value of all ones would be looped-back at this point.

The administrativeState attribute may be used by the management system to inhibit (lock) and allow (unlock) the flow of cells through the vpCTPBidirectional. However, when the vpCTPBidirectional object is configured as a segment end-point, the value of the administrativeState attribute shall not affect the ability of the vpCTPBidirectional to perform segment OAM cell processing functions.

When a VP-AIS or VP-RDI failure is detected, the vpCTPBidirectional object shall generate a communicationsAlarm notification (if the tmnCommunicationsAlarmInformationPackage is present) with the probableCause parameter value set equal to aIS or farEndReceiveFailure, respectively.

Instances of this object class may be created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively. Instances of this managed object class may also be automatically created by the managed system in response to actions performed on instances of the atmFabric object class.";

7.2.23 vpTTPBidirectional

vpTTPBidirectional MANAGED OBJECT CLASS
DERIVED FROM "ITU-T M.3100":trailTerminationPointBidirectional;
CHARACTERIZED BY
"Rec. X.721 | ISO/IEC 10165-2":administrativeStatePackage,
"ITU-T M.3100":attributeValueChangeNotificationPackage,
"ITU-T M.3100":createDeleteNotificationsPackage,

vpTTPBidirectionalPkg PACKAGE
BEHAVIOUR vpTTPBidirectionalBeh;
ATTRIBUTES
vpTTPId
GET;;;
CONDITIONAL PACKAGES
loopbackOAMCellPkg
PRESENT IF "the VPC termination point supports OAM cell Loopbacks";
REGISTERED AS { i751ObjectClass 23 };

vpTTPBidirectionalBeh BEHAVIOUR
DEFINED AS
"The vpTTPBidirectional object class is a class of managed objects that delimit Virtual Path Connections (VPCs).

An instance of this object class represents the logical point in the managed system where the end-to-end F4 flow (i.e. OAM cells with VCI=4) terminates.

The loopbackOAMCellPkg package provides the M-ACTION used to request the termination point to insert an OAM cell for downstream loopbacking and report whether or not the cell was returned within the required time.

Note that all End-to-End F4 Loopback cells with a default Loopback Location field value of all ones would be looped-back at this point.

Instances of this object class are created and deleted by the managing system using the CMIS M-CREATE and M-DELETE services, respectively. Instances of this managed object class may also be automatically created by the managed system in response to actions performed on instances of the atmFabric object class.

An instance of this object shall always point to vpCTPBidirectional managed object using UCP/DCP.";

7.2.24 vpVcPMCurrentData

vpVcPMCurrentData MANAGED OBJECT CLASS
DERIVED FROM atmCurrentData;
CHARACTERIZED BY
"ITU-T M.3100":createDeleteNotificationsPackage,
vpVcPMCurrentDataPackage PACKAGE
BEHAVIOUR vpVcPMCurrentDataBeh;
ATTRIBUTES
userCells
REPLACE-WITH-DEFAULT
DEFAULT VALUE AtmMIBMod.integerZero
GET,
lostCells
REPLACE-WITH-DEFAULT
DEFAULT VALUE AtmMIBMod.integerZero
GET,
misinsertedCells
REPLACE-WITH-DEFAULT
DEFAULT VALUE AtmMIBMod.integerZero
GET;;;
CONDITIONAL PACKAGES
feVpVcPMCurrentDataPkg
PRESENT IF "the fePMDataCollectionPkg is present in the containing instance of the bidirectionalPerformanceMonitor object and the managed system supports far-end PM data collection";
REGISTERED AS { i751ObjectClass 24 };

vpVcPMCurrentDataBeh BEHAVIOUR
DEFINED AS
"This managed object contains the current PM data collected for the connection being monitored. It includes attributes defined for the VP and VC OAM flow related to performance data collection.

If Far-End data collection is supported, this object shall contain Far-End data as well.

This object shall be automatically created whenever an instance of the bidirectionalPerformanceMonitor object class is created. Instances of this object shall be automatically deleted when the containing instance of the bidirectionalPerformanceMonitor object class is deleted.";

7.2.25 vpVcPMHistoryData

```
vpVcPMHistoryData MANAGED OBJECT CLASS
    DERIVED FROM "Rec. Q.822:1993":historyData;
    CHARACTERIZED BY
        vpVcPMHistoryDataPackage PACKAGE
            BEHAVIOUR vpVcPMHistoryDataBeh;
            ATTRIBUTES
                userCells
                    GET,
                lostCells
                    GET,
                misinsertedCells
                    GET;;;
        CONDITIONAL PACKAGES
            feVpVcPMHistoryDataPkg
                PRESENT IF "an instance supports it";
REGISTERED AS { i751ObjectClass 25 };

vpVcPMHistoryDataBeh BEHAVIOUR
    DEFINED AS
    "This managed object contains the past Performance data collected for the connection being monitored. It includes
        attributes defined for the VP and VC OAM flow related to performance data collection.

    If Far-End data collection is supported, this object shall contain Far-End data as well.";
```

7.3 Conditional Packages

7.3.1 backwardPMReportingPkg

```
backwardPMReportingPkg PACKAGE
    BEHAVIOUR backwardPMReportingPkgBeh;
    ATTRIBUTES
        backwardPMReportingMechanismActive
            INITIAL VALUE AtmMIBMod.booleanFalse
            GET;
REGISTERED AS { i751Package 1 };

backwardPMReportingPkgBeh BEHAVIOUR
    DEFINED AS
    "This package includes attributes used to indicate the far-end backward reporting activity.";
```

7.3.2 cellScramblingEnabledPkg

```
cellScramblingEnabledPkg PACKAGE
    ATTRIBUTES
        cellScramblingEnabled
            DEFAULT VALUE AtmMIBMod.booleanTrueDefault
            GET-REPLACE;
REGISTERED AS { i751Package 2 };
```

7.3.3 discardedCLP0CellsHistoryDataPkg

```
discardedCLP0CellsHistoryDataPkg PACKAGE
    ATTRIBUTES
        discardedCLP0Cells
            GET;
REGISTERED AS { i751Package 3 };
```

7.3.4 discardedCLP0CellsPkg

```
discardedCLP0CellsPkg PACKAGE
    ATTRIBUTES
        discardedCLP0Cells
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE AtmMIBMod.integerZero
            GET;
REGISTERED AS { i751Package 4 };
```

7.3.5 egressTrafficDescriptorPkg

```
egressTrafficDescriptorPkg PACKAGE
    ATTRIBUTES
        egressPeakCellRate
            GET-REPLACE,
        egressCDVTolerancePCR
            GET-REPLACE,
        egressCDVToleranceSCR
            GET-REPLACE,
        egressSustainableCellRate
            GET-REPLACE,
        egressMaxBurstSize
            GET-REPLACE;
REGISTERED AS { i751Package 5 };
```

7.3.6 fePMDataCollectionPkg

```
fePMDataCollectionPkg PACKAGE
    BEHAVIOUR fePMDataCollectionPkgBeh;
    ATTRIBUTES
        fePMDataCollectionMechanismActive
            INITIAL VALUE AtmMIBMod.booleanFalse
            GET;
REGISTERED AS { i751Package 6 };

fePMDataCollectionPkgBeh BEHAVIOUR
    DEFINED AS
    "This package includes attributes used to indicate the far-end data collection activity.";
```

7.3.7 feVpVcPMCurrentDataPkg

```
feVpVcPMCurrentDataPkg PACKAGE
    BEHAVIOUR feVpVcPMCurrentDataPkgBeh;
    ATTRIBUTES
        feUserCells
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE AtmMIBMod.integerZero
            GET,
        feLostCells
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE AtmMIBMod.integerZero
            GET,
        feMisinsertedCells
            REPLACE-WITH-DEFAULT
            DEFAULT VALUE AtmMIBMod.integerZero
            GET;
REGISTERED AS { i751Package 7 };

feVpVcPMCurrentDataPkgBeh BEHAVIOUR
    DEFINED AS
    "This package includes attributes defined for the VP and VC OAM flow related to far end performance data collection
     report.";
```

7.3.8 feVpVcPMHistoryDataPkg

```
feVpVcPMHistoryDataPkg PACKAGE
        BEHAVIOUR feVpVcPMHistoryDataPkgBeh;
        ATTRIBUTES
            feUserCells
                GET,
            feLostCells
                GET,
            feMisinsertedCells
                GET;
REGISTERED AS { i751Package 8 };

feVpVcPMHistoryDataPkgBeh BEHAVIOUR
        DEFINED AS
        "This package includes attributes defined for the VP and VC OAM flow related to far end performance data collection
         report.";
```

7.3.9 flowDirectionPackage

```
flowDirectionPackage PACKAGE
        BEHAVIOUR flowDirectionPackageBeh;
        ATTRIBUTES
            flowDirection
                GET;
REGISTERED AS { i751Package 9 };

flowDirectionPackageBeh BEHAVIOUR
        DEFINED AS
        "This package includes attributes used to indicate the flow direction.";
```

7.3.10 incomingOamCellsHistoryDataPkg

```
incomingOamCellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            incomingOamCells
                GET;
REGISTERED AS { i751Package 10 };
```

7.3.11 ingressTrafficDescriptorPkg

```
ingressTrafficDescriptorPkg PACKAGE
        ATTRIBUTES
            ingressPeakCellRate
                GET-REPLACE,
            ingressCDVTolerancePCR
                GET-REPLACE,
            ingressCDVToleranceSCR
                GET-REPLACE,
            ingressSustainableCellRate
                GET-REPLACE,
            ingressMaxBurstSize
                GET-REPLACE;
REGISTERED AS { i751Package 11 };
```

7.3.12 loopbackLocationIdentifierPkg

```
loopbackLocationIdentifierPkg PACKAGE
        ATTRIBUTES
            loopbackLocationIdentifier
                GET-REPLACE;
REGISTERED AS { i751Package 12 };
```

7.3.13 loopbackOAMCellPkg

```
loopbackOAMCellPkg PACKAGE
        ACTIONS
            loopbackOAMCell;
REGISTERED AS { i751Package 13 };
```

7.3.14 maxBandwidthPackage

```
maxBandwidthPackage PACKAGE
        ATTRIBUTES
            maxIngressBandwidth
                GET-REPLACE,
            maxEgressBandwidth
                GET-REPLACE;
REGISTERED AS { i751Package 14 };
```

7.3.15 monitoredFlowPkg

```
monitoredFlowPkg PACKAGE
        ATTRIBUTES
            monitoredFlow
                GET;
REGISTERED AS { i751Package 15 };
```

7.3.16 oamCellCountingPkg

```
oamCellCountingPkg PACKAGE
        ATTRIBUTES
            incomingOamCells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET,
            outgoingOamCells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET;
REGISTERED AS { i751Package 16 };
```

7.3.17 oamDiscardedCellsHistoryDataPkg

```
oamDiscardedCellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            oamDiscardedCells
                GET;
REGISTERED AS { i751Package 17 };
```

7.3.18 oamDiscardedCellsPkg

```
oamDiscardedCellsPkg PACKAGE
        ATTRIBUTES
            oamDiscardedCells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET;
REGISTERED AS { i751Package 18 };
```

7.3.19 oamEgressTrafficDescriptorPkg

```
oamEgressTrafficDescriptorPkg PACKAGE
        ATTRIBUTES
            oamEgressPeakCellRate
                GET-REPLACE,
            oamEgressCDVTolerance
                GET-REPLACE;
REGISTERED AS { i751Package 19 };
```

7.3.20 oamIngressTrafficDescriptorPkg

```
oamIngressTrafficDescriptorPkg PACKAGE
        ATTRIBUTES
            oamIngressPeakCellRate
                GET-REPLACE,
            oamIngressCDVTolerance
                GET-REPLACE;
REGISTERED AS { i751Package 20 };
```

7.3.21 oamSuccessfullyPassedCellsHistoryDataPkg

```
oamSuccessfullyPassedCellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            oamSuccessfullyPassedCells
                GET;
REGISTERED AS { i751Package 21 };
```

7.3.22 oamSuccessfullyPassedCellsPkg

```
oamSuccessfullyPassedCellsPkg PACKAGE
        ATTRIBUTES
            oamSuccessfullyPassedCells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET;
REGISTERED AS { i751Package 22 };
```

7.3.23 outgoingOamCellsHistoryDataPkg

```
outgoingOamCellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            outgoingOamCells
                GET;
REGISTERED AS { i751Package 23 };
```

7.3.24 qosClassesPkg

```
qosClassesPkg PACKAGE
        ATTRIBUTES
            ingressQOSClass
                GET,
            egressQOSClass
                GET;
REGISTERED AS { i751Package 24 };
```

7.3.25 successfullyPassedCLP0CellsHistoryDataPkg

```
successfullyPassedCLP0CellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            successfullyPassedCLP0Cells
                GET;
REGISTERED AS { i751Package 25 };
```

7.3.26 successfullyPassedCLP0CellsPkg

```
successfullyPassedCLP0CellsPkg PACKAGE
        ATTRIBUTES
            successfullyPassedCLP0Cells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET;
REGISTERED AS { i751Package 26 };
```

7.3.27 taggedCLP0CellsHistoryDataPkg

```
taggedCLP0CellsHistoryDataPkg PACKAGE
        ATTRIBUTES
            taggedCLP0Cells
                GET;
REGISTERED AS { i751Package 27 };
```

7.3.28 taggedCLP0CellsPkg

```
taggedCLP0CellsPkg PACKAGE
        ATTRIBUTES
            taggedCLP0Cells
                REPLACE-WITH-DEFAULT
                DEFAULT VALUE AtmMIBMod.integerZero
                GET;
REGISTERED AS { i751Package 28 };
```

7.3.29 vcLevelProfilePackage

```
vcLevelProfilePackage PACKAGE
        ATTRIBUTES
            maxNumVCIBitsNearEnd
                GET,
            maxNumVCIBitsSupported
                GET-REPLACE,
            maxNumActiveVCCsAllowed
                GET-REPLACE,
            maxNumActiveVCCsNearEnd
                GET;
REGISTERED AS { i751Package 29 };
```

7.3.30 vpLevelProfilePackage

```
vpLevelProfilePackage PACKAGE
        ATTRIBUTES
            maxNumVPIBitsNearEnd
                GET,
            maxNumVPIBitsSupported
                GET-REPLACE,
            maxNumActiveVPCsAllowed
                GET-REPLACE,
            maxNumActiveVPCsNearEnd
                GET;
REGISTERED AS { i751Package 30 };
```

7.4 Attributes

7.4.1 atmAccessProfileId

```
atmAccessProfileId ATTRIBUTE
            WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
            MATCHES FOR EQUALITY;
            BEHAVIOUR atmAccessProfileIdBeh;
REGISTERED AS { i751Attribute 1 };

atmAccessProfileIdBeh BEHAVIOUR
            DEFINED AS
            "This attribute is used to name instances of the atmAccessProfile managed object class.";
```

7.4.2 atmFabricId

```
atmFabricId ATTRIBUTE
            WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
            MATCHES FOR EQUALITY;
            BEHAVIOUR atmFabricIdBeh;
REGISTERED AS { i751Attribute 2 };

atmFabricIdBeh BEHAVIOUR
            DEFINED AS
            "This attribute is used to name instances of the atmFabric managed object class.";
```

7.4.3 backwardPMReportingMechanismActive

```
backwardPMReportingMechanismActive ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
        MATCHES FOR EQUALITY;
        BEHAVIOUR backwardPMReportingMechanismActiveBeh;
REGISTERED AS { i751Attribute 3 };

backwardPMReportingMechanismActiveBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate if backward reporting of performance data is active or not.";
```

7.4.4 cellHeaderAbnormalityType

```
cellHeaderAbnormalityType ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.CellHeaderAbnormalityType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR cellHeaderAbnormalityTypeBeh;
REGISTERED AS { i751Attribute 4 };

cellHeaderAbnormalityTypeBeh BEHAVIOUR
        DEFINED AS
            "This attribute identifies the abnormality associated with the log record. Valid values for this attribute
                are: unassigned VPI/VCI Value and Out-Of-Range VPI/VCI Value.";
```

7.4.5 cellScramblingEnabled

```
cellScramblingEnabled ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
        MATCHES FOR EQUALITY;
        BEHAVIOUR cellScramblingEnabledBeh;
REGISTERED AS { i751Attribute 5 };

cellScramblingEnabledBeh BEHAVIOUR
        DEFINED AS
            "This attribute identifies whether or not ATM cell scrambling is being performed over the ATM
                interface. A value of TRUE (default) is used to indicate that cell scrambling is being
                performed.";
```

7.4.6 continuityMonitorId

```
continuityMonitorId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR continuityMonitorIdBeh;
REGISTERED AS { i751Attribute 6 };

continuityMonitorIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used for naming bidirectionalContinuityMonitor objects.";
```

7.4.7 discardedCells

```
discardedCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR discardedCellsBeh;
REGISTERED AS { i751Attribute 7 };

discardedCellsBeh BEHAVIOUR
        DEFINED AS
            "This attribute provides a count of the number of ATM cells that were discarded due to UPC/NPC
                policing of the combined high and low cell loss priority traffic.";
```

7.4.8 discardedCellsHECViolation

```
discardedCellsHECViolation ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR discardedCellsHECViolationBeh;
REGISTERED AS { i751Attribute 8 };
```

discardedCellsHECViolationBeh BEHAVIOUR
DEFINED AS
"This attribute provides a count of the number of cells discarded due to uncorrectable header bit errors.";

7.4.9 discardedCellsInvalidHeader

discardedCellsInvalidHeader ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR discardedCellsInvalidHeaderBeh;
REGISTERED AS { i751Attribute 9 };

discardedCellsInvalidHeaderBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the number of ATM cells discarded due to header content errors.";

7.4.10 discardedCLP0Cells

discardedCLP0Cells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR discardedCLP0CellsBeh;
REGISTERED AS { i751Attribute 10 };

discardedCLP0CellsBeh BEHAVIOUR
DEFINED AS
"This attribute provides a count of the number of cells with CLP=0 that were discarded due to UPC/NPC policing of high priority (CLP=0) only traffic.";

7.4.11 egressCDVTolerancePCR

egressCDVTolerancePCR ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR egressCDVTolerancePCRBeh;
REGISTERED AS { i751Attribute 11 };

egressCDVTolerancePCRBeh BEHAVIOUR
DEFINED AS
"This attribute represents the egress (with respect to the managed system) CDV Tolerance related to Peak Cell Rate (PCR) assigned to the VPL or VCL being terminated.";

7.4.12 egressCDVToleranceSCR

egressCDVToleranceSCR ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR egressCDVToleranceSCRBeh;
REGISTERED AS { i751Attribute 12 };

egressCDVToleranceSCRBeh BEHAVIOUR
DEFINED AS
"This attribute represents the egress (with respect to the managed system) CDV Tolerance related to Sustainable Cell Rate (SCR) assigned to the VPL or VCL being terminated.";

7.4.13 egressMaxBurstSize

egressMaxBurstSize ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.MaxBurstSize;
MATCHES FOR EQUALITY;
BEHAVIOUR egressMaxBurstSizeBeh;
REGISTERED AS { i751Attribute 13 };

egressMaxBurstSizeBeh BEHAVIOUR
DEFINED AS
"This attribute represents the egress (with respect to the managed system) maximum burst size that has been assigned to the VP or VC link being terminated.";

7.4.14 egressPeakCellRate

```
egressPeakCellRate ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.PeakCellRate;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR egressPeakCellRateBeh;
REGISTERED AS { i751Attribute 14 };

egressPeakCellRateBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate the peak cell rate assigned or reserved in the egress (with respect to
                the managed element) direction of transmission across the VP or VC link being terminated.";
```

7.4.15 egressQOSClass

```
egressQOSClass ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.QosClass;
        MATCHES FOR EQUALITY;
        BEHAVIOUR egressQOSClassBeh;
REGISTERED AS { i751Attribute 15 };

egressQOSClassBeh BEHAVIOUR
        DEFINED AS
            "This attribute identifies the Quality Of Service (QOS) class assigned to the VPL or VCL in the
                egress (with respect to the managed element) direction of cell transmission. Valid values for this
                attribute are: Class 0, Class 1, Class 2, Class 3, and Class 4.";
```

7.4.16 egressSustainableCellRate

```
egressSustainableCellRate ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.SustainableCellRate;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR egressSustainableCellRateBeh;
REGISTERED AS { i751Attribute 16};

egressSustainableCellRateBeh BEHAVIOUR
        DEFINED AS
            "This traffic descriptor represents the egress (with respect to the managed system) sustainable cell
                rate assigned to the link being terminated.";
```

7.4.17 erroredCellsHECViolation

```
erroredCellsHECViolation ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR erroredCellsHECViolationBeh;
REGISTERED AS { i751Attribute 17 };

erroredCellsHECViolationBeh BEHAVIOUR
        DEFINED AS
            "This attribute provides a count of the number of cells with header bit errors.";
```

7.4.18 feLostCells

```
feLostCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR feLostCellsBehaviour;
REGISTERED AS { i751Attribute 18 };

feLostCellsBehaviour BEHAVIOUR
        DEFINED AS
            "This attribute countains the count of lost cells that were detected by the far-end terminal.";
```

7.4.19 feMisinsertedCells

```
feMisinsertedCells ATTRIBUTE

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR feMisinsertedCellsBehaviour;
REGISTERED AS { i751Attribute 19 };
```

feMisinsertedCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute countains the count of misinserted cells that were detected by the far end terminal.";

7.4.20 fePMDataCollectionMechanismActive

fePMDataCollectionMechanismActive ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
MATCHES FOR EQUALITY;
BEHAVIOUR fePMDataCollectionMechanismActiveBeh;
REGISTERED AS { i751Attribute 20 };

fePMDataCollectionMechanismActiveBeh BEHAVIOUR
DEFINED AS
"This attribute is used to indicate if far-end performance data collection is active or not.";

7.4.21 feUserCells

feUserCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR feUserCellsBehaviour;
REGISTERED AS { i751Attribute 21 };

feUserCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute countains the count of incoming user information cells processed on the termination point being monitored by the far-end terminal.";

7.4.22 flowDirection

flowDirection ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.FlowDirection;
MATCHES FOR EQUALITY;
BEHAVIOUR flowDirectionBeh;
REGISTERED AS { i751Attribute 22 };

flowDirectionBeh BEHAVIOUR
DEFINED AS
"The flowDirection attribute indicates in which direction the monitoring is made, in relation to the TP by which the monitor object is named. The value *outOfSwitch* means that the monitored flow extends from TP toward the network. The value *inToSwitch* means that monitored flow extends from TP into the cross-connect. If both flows (inToSwitch and outOfSwitch) need to be monitored, then two monitor objects should be created.";

7.4.23 inAverageMonitoringBlockSize

inAverageMonitoringBlockSize ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.AverageMonitoringBlockSize;
MATCHES FOR EQUALITY;
BEHAVIOUR inAverageMonitoringBlockSizeBeh;
REGISTERED AS { i751Attribute 23 };

inAverageMonitoringBlockSizeBeh BEHAVIOUR
DEFINED AS
"This attribute contains the average monitoring block size, associated with the incoming ATM cell flow along a VPC/VCC. The inAverageMonitoringBlockSize may be set to a length of 128, 256, 512 or 1024 cells.";

7.4.24 incomingCells

incomingCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR incomingCellsBehaviour;
REGISTERED AS { i751Attribute 24 };

incomingCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute contains the number of incoming cells.";

7.4.25 incomingOamCells

incomingOamCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR incomingOamCellsBehaviour;
REGISTERED AS { i751Attribute 25 };

incomingOamCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute contains the number of incoming OAM cells.";

7.4.26 ingressCDVTolerancePCR

ingressCDVTolerancePCR ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR ingressCDVTolerancePCRBeh;
REGISTERED AS { i751Attribute 26 };

ingressCDVTolerancePCRBeh BEHAVIOUR
DEFINED AS
"This attribute represents the ingress (with respect to the managed system) CDV Tolerance related to Peak Cell Rate (PCR) assigned to the VPL or VCL being terminated.";

7.4.27 ingressCDVToleranceSCR

ingressCDVToleranceSCR ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR ingressCDVToleranceSCRBeh;
REGISTERED AS { i751Attribute 27 };

ingressCDVToleranceSCRBeh BEHAVIOUR
DEFINED AS
"This attribute represents the ingress (with respect to the managed system) CDV Tolerance related to Sustainable Cell Rate (SCR) assigned to the VPL or VCL being terminated.";

7.4.28 ingressMaxBurstSize

ingressMaxBurstSize ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.MaxBurstSize;
MATCHES FOR EQUALITY;
BEHAVIOUR ingressMaxBurstSizeBeh;
REGISTERED AS { i751Attribute 28 };

ingressMaxBurstSizeBeh BEHAVIOUR
DEFINED AS
"This attribute represents the ingress (with respect to the managed system) maximum burst size that has been assigned to the VP or VC link being terminated.";

7.4.29 ingressPeakCellRate

ingressPeakCellRate ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.PeakCellRate;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR ingressPeakCellRateBeh;
REGISTERED AS { i751Attribute 29 };

ingressPeakCellRateBeh BEHAVIOUR
DEFINED AS
"This attribute is used to indicate the peak cell rate assigned or reserved in the ingress (with respect to the managed system) direction of transmission across the VP or VC link being terminated.";

7.4.30 ingressQOSClass

```
ingressQOSClass ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.QosClass;
        MATCHES FOR EQUALITY;
        BEHAVIOUR ingressQOSClassBeh;
REGISTERED AS { i751Attribute 30};

ingressQOSClassBeh BEHAVIOUR
        DEFINED AS
            "This attribute identifies the Quality of Service (QOS) class assigned to the VPL or VCL in the
                gingress (with respect to the managed system) direction of cell transmission. Valid values for
                this attribute are: Class 0, Class 1, Class 2, Class 3, and Class 4.";
```

7.4.31 ingressSustainableCellRate

```
ingressSustainableCellRate ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.SustainableCellRate;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR ingressSustainableCellRateBeh;
REGISTERED AS { i751Attribute 31 };

ingressSustainableCellRateBeh BEHAVIOUR
        DEFINED AS
            "This traffic descriptor represents the ingress (with respect to the managed system) sustainable cell
                rate assigned to the link being terminated.";
```

7.4.32 interfacePointer

```
interfacePointer ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.InterfacePointer;
        MATCHES FOR EQUALITY;
        BEHAVIOUR interfacePointerBeh;
REGISTERED AS { i751Attribute 32 };

interfacePointerBeh BEHAVIOUR
        DEFINED AS
            "This attribute identifies the object instance Id of the uni, interNNI, or intraNNI object with which the
                cell header abnormality is associated.";
```

7.4.33 interNNIId

```
interNNIId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR interNNIIdBeh;
REGISTERED AS { i751Attribute 33 };

interNNIIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to name instances of the interNNI managed object class.";
```

7.4.34 intraNNIId

```
intraNNIId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR intraNNIIdBeh;
REGISTERED AS { i751Attribute 34 };

intraNNIIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to name instances of the intraNNI managed object class.";
```

7.4.35 loopbackLocationIdentifier

```
loopbackLocationIdentifier ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.OctetString;
        MATCHES FOR EQUALITY;
        BEHAVIOUR loopbackLocationIdentifierBeh;
REGISTERED AS { i751Attribute 35 };
```

loopbackLocationIdentifierBeh BEHAVIOUR
DEFINED AS
"This attribute provides a read/write code used for OAM cell loopback purposes. Incoming OAM Loopback cells with a Loopback Location field value that matches the value of the loopbackLocationIdentifier attribute shall be looped-back over the ATM interface.";

7.4.36 lostCells

lostCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR lostCellsBehaviour;
REGISTERED AS { i751Attribute 36 };

lostCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute countains the count of detected lost cells.";

7.4.37 maxEgressBandwidth

maxEgressBandwidth ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxEgressBandwidthBeh;
REGISTERED AS { i751Attribute 37 };

maxEgressBandwidthBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum egress bandwidth for the ATM interface that is available to the managed system for support of non-provisioned ATM cross-connections.";

7.4.38 maxIngressBandwidth

maxIngressBandwidth ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxIngressBandwidthBeh;
REGISTERED AS { i751Attribute 38 };

maxIngressBandwidthBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum ingress bandwidth for the ATM interface that is available to the managed system for support of non-provisioned ATM cross-connections.";

7.4.39 maxNumActiveVCCsAllowed

maxNumActiveVCCsAllowed ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumActiveVCCsAllowedBeh;
REGISTERED AS { i751Attribute 39 };

maxNumActiveVCCsAllowedBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of concurrently active Virtual Channel Connections (VCCs) that the entity being profiled (e.g. tcAdaptorTTPBidirectional or vpTTPBidirectional) has been configured to support. The managing system is not required to supply this attribute value in the M-CREATE request of an instance of atmAccessProfile. If the attribute is not supplied by the managing system, then the attribute value shall be determined by the managed system, and reported to the managing system in the M-CREATE response. If the value of the attribute supplied by the managing system is greater than the managed system capability, the managed system will set the value.";

7.4.40 maxNumActiveVCCsNearEnd

maxNumActiveVCCsNearEnd ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumActiveVCCsNearEndBeh;
REGISTERED AS { i751Attribute 40 };

maxNumActiveVCCsNearEndBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of concurrently active Virtual Channel Connections (VCCs) that can be locally supported by the entity being profiled (e.g. tcAdaptorTTPBidirectional or vpTTPBidirectional). The managing system shall not supply this attribute value in the M-CREATE request. The attribute value shall be determined by the managed system according to its capability, and shall be reported in the M-CREATE response.";

7.4.41 maxNumActiveVPCsAllowed

maxNumActiveVPCsAllowed ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumActiveVPCsAllowedBeh;
REGISTERED AS { i751Attribute 41 };

maxNumActiveVPCsAllowedBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of concurrently active Virtual Path Connections (VPCs) that the interface has been configured to support. The managing system is not required to supply this attribute value in the M-CREATE request of an instance of atmAccessProfile. If the attribute is not supplied by the managing system, then the attribute value shall be determined by the managed system, and reported to the managing system in the M-CREATE response. If the value of the attribute supplied by the managing system is greater than the managed system capability, the managed system will set the value.";

7.4.42 maxNumActiveVPCsNearEnd

maxNumActiveVPCsNearEnd ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumActiveVPCsNearEndBeh;
REGISTERED AS { i751Attribute 42 };

maxNumActiveVPCsNearEndBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of concurrently active Virtual Path Connections (VPCs) that can be locally supported by the equipment terminating the UNI, inter-NNI, or intra-NNI. The managing system shall not supply this attribute value in the M-CREATE request. The attribute value shall be determined by the managed system according to its capability, and shall be reported in the M-CREATE response.";

7.4.43 maxNumVCIBitsNearEnd

maxNumVCIBitsNearEnd ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumVCIBitsNearEndBeh;
REGISTERED AS { i751Attribute 43 };

maxNumVCIBitsNearEndBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of contiguous VCI bits, starting from the least significant bit, that can be locally supported by the equipment terminating the UNI, inter-NNI, or intra-NNI. The default value for this attribute is the entire VCI field length of the ATM cell, i.e. 16 bits. The managing system shall not supply this attribute value in the M-CREATE request. The attribute value shall be determined by the managed system according to its capability, and shall be reported in the M-CREATE response.";

7.4.44 maxNumVCIBitsSupported

maxNumVCIBitsSupported ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumVCIBitsSupportedBeh;
REGISTERED AS { i751Attribute 44 };

maxNumVCIBitsSupportedBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of contiguous VCI bits, starting from the least significant bit, that may be used over the UNI, inter-NNI, or intra-NNI. The managing system is not required to supply this attribute value in the M-CREATE request of an instance of atmAccessProfile. If the attribute is not supplied by the managing system, then the attribute value shall be determined by the managed system, and reported to the managing system in the M-CREATE response. If the value of the attribute supplied by the managing system is greater than the managed system capability, the managed system will set the value.";

7.4.45 maxNumVPIBitsNearEnd

maxNumVPIBitsNearEnd ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumVPIBitsNearEndBeh;
REGISTERED AS { i751Attribute 45 };

maxNumVPIBitsNearEndBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of contiguous VPI bits, starting from the least significant bit, that can be locally supported by the equipment terminating the UNI, inter-NNI, or intra-NNI. The default value for this attribute is the entire VPI field length of the ATM cell (8 bits for UNIs and 12 bits for NNIs). The managing system shall not supply this attribute value in the M-CREATE request. The attribute value shall be determined by the managed system according to its capability, and shall be reported in the M-CREATE response.";

7.4.46 maxNumVPIBitsSupported

maxNumVPIBitsSupported ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR maxNumVPIBitsSupportedBeh;
REGISTERED AS { i751Attribute 46 };

maxNumVPIBitsSupportedBeh BEHAVIOUR
DEFINED AS
"This attribute identifies the maximum number of contiguous VPI bits, starting from the least significant bit, that may be used over the UNI, inter-NNI, or intra-NNI.The managing system is not required to supply this attribute value in the M-CREATE request of an instance of atmAccessProfile. If the attribute is not supplied by the managing system, then the attribute value shall be determined by the managed system, and reported to the managing system in the M-CREATE response. If the value of the attribute supplied by the managing system is greater than the managed system capability, the managed system will set the value.";

7.4.47 misinsertedCells

misinsertedCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR misinsertedCellsBehaviour;
REGISTERED AS { i751Attribute 47 };

misinsertedCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute countains the count of detected misinserted cells.";

7.4.48 monitoredFlow

monitoredFlow ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.OamCellType;
MATCHES FOR EQUALITY;
BEHAVIOUR monitoredFlowBeh;
REGISTERED AS { i751Attribute 48 };

monitoredFlowBeh BEHAVIOUR
DEFINED AS
"This attribute is used for indicating if the non-intrusive Performance Monitoring is segment or end to end.";

7.4.49 oamDiscardedCells

```
oamDiscardedCells ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR oamDiscardedCellsBeh;
REGISTERED AS { i751Attribute 49 };

oamDiscardedCellsBeh BEHAVIOUR
        DEFINED AS
            "This attribute Contains the number of OAM cells discarded by the UPC or NPC functions.";
```

7.4.50 oamEgressCDVTolerance

```
oamEgressCDVTolerance ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
        MATCHES FOR EQUALITY,ORDERING;
        BEHAVIOUR oamEgressCDVToleranceBeh;
REGISTERED AS { i751Attribute 50 };

oamEgressCDVToleranceBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate the oam cell delay variation assigned or reserved in the oam egress
                direction.";
```

7.4.51 oamEgressPeakCellRate

```
oamEgressPeakCellRate ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.PeakCellRate;
        MATCHES FOR EQUALITY,ORDERING;
        BEHAVIOUR oamEgressPeakCellRateBeh;
REGISTERED AS { i751Attribute 51 };

oamEgressPeakCellRateBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate the oam peak cell rate assigned or reserved in the oam egress
                direction.";
```

7.4.52 oamIngressCDVTolerance

```
oamIngressCDVTolerance ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.CDVTolerance;
        MATCHES FOR EQUALITY,ORDERING;
        BEHAVIOUR oamIngressCDVToleranceBeh;
REGISTERED AS { i751Attribute 52 };

oamIngressCDVToleranceBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate the oam cell delay variation assigned or reserved in the oam ingress
                direction.";
```

7.4.53 oamIngressPeakCellRate

```
oamIngressPeakCellRate ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.PeakCellRate;
        MATCHES FOR EQUALITY,ORDERING;
        BEHAVIOUR oamIngressPeakCellRateBeh;
REGISTERED AS { i751Attribute 53 };

oamIngressPeakCellRateBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used to indicate the oam peak cell rate assigned or reserved in the oam ingress
                direction.";
```

7.4.54 oamSuccessfullyPassedCells

```
oamSuccessfullyPassedCells ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR oamSuccessfullyPassedCellsBeh;
REGISTERED AS { i751Attribute 54 };

oamSuccessfullyPassedCellsBeh BEHAVIOUR
        DEFINED AS
            "This attribute Contains the number of OAM cells successfully passed through the UPC or NPC
                functions.";
```

7.4.55 outAverageMonitoringBlockSize

```
outAverageMonitoringBlockSize ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.AverageMonitoringBlockSize;
        MATCHES FOR EQUALITY;
        BEHAVIOUR outAverageMonitoringBlockSizeBeh;
REGISTERED AS { i751Attribute 55 };

outAverageMonitoringBlockSizeBeh BEHAVIOUR
        DEFINED AS
            "This attribute contains the average monitoring block size, associated with the outgoing direction of a
                monitored ATM cell flow along a VPC/VCC. The outAverageMonitoringBlockSize may be set
                to a length of 128, 256, 512 or 1024 cells. This attribute is automatically set as a result of the
                VPC performance monitoring activation.";
```

7.4.56 outgoingCells

```
outgoingCells ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR outgoingCellsBehaviour;
REGISTERED AS { i751Attribute 56 };

outgoingCellsBehaviour BEHAVIOUR
        DEFINED AS
            "This attribute contains the number of outgoing cells.";
```

7.4.57 outgoingOamCells

```
outgoingOamCells ATTRIBUTE
        DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
        BEHAVIOUR outgoingOamCellsBehaviour;
REGISTERED AS { i751Attribute 57 };

outgoingOamCellsBehaviour BEHAVIOUR
        DEFINED AS
            "This attribute contains the number of outgoing OAM cells.";
```

7.4.58 performanceMonitorId

```
performanceMonitorId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR performanceMonitorIdBeh;
REGISTERED AS { i751Attribute 58 };

performanceMonitorIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used for naming bidirectionalPerformanceMonitor objects.";
```

7.4.59 segmentEndPoint

```
segmentEndPoint ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
        MATCHES FOR EQUALITY;
        BEHAVIOUR segmentEndPointBeh;
REGISTERED AS { i751Attribute 59 };

segmentEndPointBeh BEHAVIOUR
        DEFINED AS
            "This boolean attribute indicates whether the vpCTPBidirectional object instance or
                vcCTPBidirectional object instance has been configured to represent an end-point of a VPC or
                VCC Segment, respectively.";
```

7.4.60 sinkCCMechanismActive

```
sinkCCMechanismActive ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
        MATCHES FOR EQUALITY;
        BEHAVIOUR sinkCCMechanismActiveBeh;
REGISTERED AS { i751Attribute 60 };
```

sinkCCMechanismActiveBeh BEHAVIOUR
DEFINED AS
"This attribute indicates whether the processing of the CC OAM flow is active.";

7.4.61 sinkPMMechanismActive

sinkPMMechanismActive ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
MATCHES FOR EQUALITY;
BEHAVIOUR sinkPMMechanismActiveBeh;
REGISTERED AS { i751Attribute 61 };

sinkPMMechanismActiveBeh BEHAVIOUR
DEFINED AS
"This attribute indicates whether the processing of the PM OAM flow is active.";

7.4.62 sourceCCMechanismActive

sourceCCMechanismActive ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
MATCHES FOR EQUALITY;
BEHAVIOUR sourceCCMechanismActiveBeh;
REGISTERED AS { i751Attribute 62 };

sourceCCMechanismActiveBeh BEHAVIOUR
DEFINED AS
"This attribute indicates whether the generation of the CC OAM flow is active.";

7.4.63 sourcePMMechanismActive

sourcePMMechanismActive ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.Boolean;
MATCHES FOR EQUALITY;
BEHAVIOUR sourcePMMechanismActiveBeh;
REGISTERED AS { i751Attribute 63 };

sourcePMMechanismActiveBeh BEHAVIOUR
DEFINED AS
"This attribute indicates whether the generation of the PM OAM flow is active.";

7.4.64 successfullyPassedCells

successfullyPassedCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR successfullyPassedCellsBeh;
REGISTERED AS { i751Attribute 64 };

successfullyPassedCellsBeh BEHAVIOUR
DEFINED AS
"This attribute represents the number of ATM cells that where received and successfully passed (i.e. not discarded) by the UPC/NPC function after performing policing functions on the combined high and low cell loss priority traffic.";

7.4.65 successfullyPassedCLP0Cells

successfullyPassedCLP0Cells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR successfullyPassedCLP0CellsBeh;
REGISTERED AS { i751Attribute 65 };

successfullyPassedCLP0CellsBeh BEHAVIOUR
DEFINED AS
"This attribute represents the number of ATM cells that where received and successfully passed (i.e. not discarded) by the UPC/NPC function after performing policing functions on the high priority (CLP=0) traffic.";

7.4.66 taggedCLP0Cells

taggedCLP0Cells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR taggedCLP0CellsBeh;
REGISTERED AS { i751Attribute 66 };

taggedCLP0CellsBeh BEHAVIOUR
DEFINED AS
"This attribute provides a count of the number of cells with CLP=0 that were tagged (i.e. CLP reset to 1) by the UPC or NPC function.";

7.4.67 tcTTPId

tcTTPId ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
MATCHES FOR EQUALITY;
BEHAVIOUR tcTTPIdBeh;
REGISTERED AS { i751Attribute 67 };

tcTTPIdBeh BEHAVIOUR
DEFINED AS
"This attribute is used for naming instances of the tcAdaptorTTPBidirectional managed object class.";

7.4.68 underlyingTTPPointer

underlyingTTPPointer ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.PointerOrNull;
MATCHES FOR EQUALITY;
BEHAVIOUR underlyingTTPPointerBeh;
REGISTERED AS { i751Attribute 68 };

underlyingTTPPointerBeh BEHAVIOUR
DEFINED AS
"This attribute provides a pointer relationship to the tcAdaptorTTPBidirectional object or vpTTPBidirectional object that provides the underlying ATM transport for the interface.";

7.4.69 uniId

uniId ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
MATCHES FOR EQUALITY;
BEHAVIOUR uniIdBeh;
REGISTERED AS { i751Attribute 69 };

uniIdBeh BEHAVIOUR
DEFINED AS
"This attribute is used for naming instances of the uni managed object class.";

7.4.70 userCells

userCells ATTRIBUTE
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2":counter;
BEHAVIOUR userCellsBehaviour;
REGISTERED AS { i751Attribute 70 };

userCellsBehaviour BEHAVIOUR
DEFINED AS
"This attribute countains the count of incoming user information cells processed on the termination point being monitored.";

7.4.71 vcCTPId

vcCTPId ATTRIBUTE
WITH ATTRIBUTE SYNTAX AtmMIBMod.VciValue;
MATCHES FOR EQUALITY;
BEHAVIOUR vcCTPIdBeh;
REGISTERED AS { i751Attribute 71 };

vcCTPIdBeh BEHAVIOUR
DEFINED AS
"This attribute is used for naming instances of the vcCTPBidirectional managed object class. The value of this attribute shall be set equal to the VCI value of the Virtual Channel Link (VCL) being terminated.";

7.4.72 vciValue

```
vciValue ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR vciValueBeh;
REGISTERED AS { i751Attribute 72 };

vciValueBeh BEHAVIOUR
        DEFINED AS
            "This attribute represents the VCI Value in the header of the discarded ATM cell.";
```

7.4.73 vcTTPId

```
vcTTPId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR vcTTPIdBeh;
REGISTERED AS { i751Attribute 73 };

vcTTPIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used for naming instances of the vcTTPBidirectional managed object class.";
```

7.4.74 vpCTPId

```
vpCTPId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.VpiValue;
        MATCHES FOR EQUALITY;
        BEHAVIOUR vpCTPIdBeh;
REGISTERED AS { i751Attribute 74 };

vpCTPIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used for naming instances of the vpCTPBidirectional managed object class. This
                attribute shall be set equal to the VPI value of the Virtual Path Link (VPL) being terminated.";
```

7.4.75 vpiValue

```
vpiValue ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.Integer;
        MATCHES FOR EQUALITY, ORDERING;
        BEHAVIOUR vpiValueBeh;
REGISTERED AS { i751Attribute 75 };

vpiValueBeh BEHAVIOUR
        DEFINED AS
            "This attribute represents the VPI value in the header of the discarded ATM cell.";
```

7.4.76 vpTTPId

```
vpTTPId ATTRIBUTE
        WITH ATTRIBUTE SYNTAX AtmMIBMod.NameType;
        MATCHES FOR EQUALITY;
        BEHAVIOUR vpTTPIdBeh;
REGISTERED AS { i751Attribute 76 };

vpTTPIdBeh BEHAVIOUR
        DEFINED AS
            "This attribute is used for naming instances of the vpTTPBidirectional managed object class.";
```

7.5 Actions

7.5.1 Connect

```
connect ACTION
        BEHAVIOUR connectBeh;
        MODE CONFIRMED;
        WITH INFORMATION SYNTAX AtmMIBMod.ConnectInformation;
        WITH REPLY SYNTAX AtmMIBMod.ConnectReply;
REGISTERED AS { i751Action 1 };
```

connectBeh BEHAVIOUR
DEFINED AS
"This action is used to establish a point-to-point ATM connection two between termination points. The termination points to be connected can be identified explicitly by specifying the associated vcCTPBidirectional object or vpCTPBidirectional object, or by specifying the charcteristics of each termination point. Multiple point-to-point connections may be requested with a single connect ACTION.

If a valid end point descriptor is provided and the connect request can be successfully carried out, the managed system would reserve the necessary resources such as the VPI and/or VCI value, and automatically create the necessary VP/VC termination points (e.g. the VP CTPs, VP TTPs, and VC CTPs) for the cross-connection.

The result, if successful, always returns an explicit list of termination points.

Successful execution of this action would result in the creation of an instance of the atmCrossConnection object. This cross-connection object has the fromTermination and toTermination attributes pointing to the two termination points. The administrativeState attribute in the cross-connection object is initialized according to the values provided in the action request information. If the administrativeState parameter is omitted, the administrative state will be set to "unlocked".

If the administrativeState in the atmCrossConnection object is unlocked, the upstreamConnectivityPointer and downstreamConnectivityPointer in the two termination points are set to the distinguished name of the (peer) termination point to which it is connected. Also, the crossConnectionObjectPointer in the termination points shall point to the atmCrossConnection object.

This action will fail if any of the termination points specified are already involved in a cross-connection, any of the termination point descriptors specified cannot be satisfied, or the two termination points do not have compatible termination point descriptors. This action will also fail if any of the CTPs to be cross-connected have non-NULL values for the upStreamConnectivityPointer and downStreamConnectivityPointer attributes.";

7.5.2 controlCC

controlCC ACTION
BEHAVIOUR controlCCBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX AtmMIBMod.ControlCCInformation;
WITH REPLY SYNTAX AtmMIBMod.ControlCCResult;
REGISTERED AS { i751Action 2 };

controlCCBeh BEHAVIOUR
DEFINED AS
"This action is used to request both the activation and deactivation of the Continuity Check OAM cell generation and processing procedures at the termination points upon which the continuity check function is performed, i.e. the VPC or VPC/VCC Segment.

Since it applies to bidirectionalContinuityMonitor objects, this action may, eg, requests the activation one way and the deactivation the other way.";

7.5.3 controlPM

controlPM ACTION
BEHAVIOUR controlPMBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX AtmMIBMod.ControlPMInformation;
WITH REPLY SYNTAX AtmMIBMod.ControlPMResult;
REGISTERED AS { i751Action 3 };

controlPMBeh BEHAVIOUR
DEFINED AS
"This action is used to request both the activation and deactivation of the Performance Monitoring OAM cell generation and processing procedures at the termination point upon which the performance monitoring function is performed, i.e. the VPC or VPC/VCC Segment. Since it applies to bidirectionalPerformancelMonitor objects, this action may, e.g. requests the activation one way and the deactivation the other way.";

7.5.4 Disconnect

disconnect ACTION
BEHAVIOUR disconnectBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX AtmMIBMod.DisconnectInformation;
WITH REPLY SYNTAX AtmMIBMod.DisconnectResult;
REGISTERED AS { i751Action 4 };

disconnectBeh BEHAVIOUR
DEFINED AS
"This action is used to take down a point-to-point cross-connection. The connection to be taken down is specified by identifying a termination point of the connection. The other termination point of the point-to-point connection is implicitly disconnected as well and the cross-connection object is deleted. The connectivity pointers in the disconnected termination points will be set to NULL as a result of this action. Disconnection of multiple point-to-point connections can be requested by providing multiple CTP object instances in the DisconnectInformation. Each component in the DisconnectResult sequence provides the disconnection result for the corresponding components of the DisconnectInformation sequence.

This action shall not result in the deletion of any of the termination point objects, even if one or both of the termination point objects were created as a result of the connect action.";

7.5.5 loopbackOAMCell

loopbackOAMCell ACTION
BEHAVIOUR loopbackOAMCellBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX AtmMIBMod.LoopbackOAMCellInfo;
WITH REPLY SYNTAX AtmMIBMod.LoopbackOAMCellReply;
REGISTERED AS { i751Action 5 };

loopbackOAMCellBeh BEHAVIOUR
DEFINED AS
"This action is used to request a vpCTPBidirectional, vcCTPBidirectional, vpTTPBidirectional, or vcTTPBidirectional object to insert (in the outgoing direction) a loopback OAM cell into the ATM cell stream and verify its return.

Supplied along with this action is the loopbackLocation information. This information identifies the downstream vpCTPBidirectional, vcCTPBidirectional, vpTTPBidirectional, or vcTTPBidirectional object instance responsible for looping back the OAM cell. The loopbackLocation value of TRUE-NULL (default) can be used to request the end-point of the ATM connection or connection segment to loopback the OAM cell. Also supplied with this information is an indication as to whether or not the OAM Loopback Cell to be inserted shall be of the segment type or of the end-to-end type.";

7.6 Notifications

No notifications have been defined.

7.7 Parameter definitions

No parameters have been defined.

7.8 Name-Bindings

7.8.1 atmAccessProfile-tcAdaptorTTPBidirectional

atmAccessProfile-tcAdaptorTTPBidirectional NAME BINDING
SUBORDINATE OBJECT CLASS atmAccessProfile AND SUBCLASSES;
NAMED BY SUPERIOR OBJECT CLASS tcAdaptorTTPBidirectional AND SUBCLASSES;
WITH ATTRIBUTE atmAccessProfileId;
CREATE
WITH-REFERENCE-OBJECT,
WITH-AUTOMATIC-INSTANCE-NAMING;
DELETE
DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 1 };

7.8.2 atmAccessProfile-vpTTPBidirectional

```
atmAccessProfile-vpTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS atmAccessProfile AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vpTTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE atmAccessProfileId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 2 };
```

7.8.3 atmCrossConnection-atmFabric

```
atmCrossConnection-atmFabric NAME BINDING
        SUBORDINATE OBJECT CLASS atmCrossConnection AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS atmFabric AND SUBCLASSES;
        WITH ATTRIBUTE "ITU-T M.3100":crossConnectionId;
        DELETE
            ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 3 };
```

7.8.4 atmFabric-managedElementR1

```
atmFabric-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS atmFabric AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1
        AND SUBCLASSES;
        WITH ATTRIBUTE atmFabricId;
REGISTERED AS { i751NameBinding 4 };
```

7.8.5 atmTrafficLoadCurrentData-interNNI

```
atmTrafficLoadCurrentData-interNNI NAME BINDING
        SUBORDINATE OBJECT CLASS atmTrafficLoadCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS interNNI AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 5 };
```

7.8.6 atmTrafficLoadCurrentData-intraNNI

```
atmTrafficLoadCurrentData-intraNNI NAME BINDING
        SUBORDINATE OBJECT CLASS atmTrafficLoadCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS intraNNI AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 6 };
```

7.8.7 atmTrafficLoadCurrentData-uni

```
atmTrafficLoadCurrentData-uni NAME BINDING
        SUBORDINATE OBJECT CLASS atmTrafficLoadCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS uni AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 7 };
```

7.8.8 atmTrafficLoadCurrentData-vcCTPBidirectional

```
atmTrafficLoadCurrentData-vcCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS atmTrafficLoadCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vcCTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 8 };
```

7.8.9 atmTrafficLoadCurrentData-vpCTPBidirectional

```
atmTrafficLoadCurrentData-vpCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS atmTrafficLoadCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vpCTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 9 };
```

7.8.10 bidirectionalContinuityMonitor-vcCTPBidirectional

```
bidirectionalContinuityMonitor-vcCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalContinuityMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vcCTPBidirectional;
        WITH ATTRIBUTE continuityMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 10 };
```

7.8.11 bidirectionalContinuityMonitor-vcTTPBidirectional

```
bidirectionalContinuityMonitor-vcTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalContinuityMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vcTTPBidirectional;
        WITH ATTRIBUTE continuityMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 11 };
```

7.8.12 bidirectionalContinuityMonitor-vpCTPBidirectional

```
bidirectionalContinuityMonitor-vpCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalContinuityMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vpCTPBidirectional;
        WITH ATTRIBUTE continuityMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 12 };
```

7.8.13 bidirectionalContinuityMonitor-vpTTPBidirectional

```
bidirectionalContinuityMonitor-vpTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalContinuityMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vpTTPBidirectional;
        WITH ATTRIBUTE continuityMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 13 };
```

7.8.14 bidirectionalPerformanceMonitor-vcCTPBidirectional

```
bidirectionalPerformanceMonitor-vcCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalPerformanceMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vcCTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE performanceMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 14 };
```

7.8.15 bidirectionalPerformanceMonitor-vcTTPBidirectional

```
bidirectionalPerformanceMonitor-vcTTPBidirectional  NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalPerformanceMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vcTTPBidirectional   AND SUBCLASSES;
        WITH ATTRIBUTE performanceMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;

REGISTERED AS { i751NameBinding 15 };
```

7.8.16 bidirectionalPerformanceMonitor-vpCTPBidirectional

```
bidirectionalPerformanceMonitor-vpCTPBidirectional  NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalPerformanceMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vpCTPBidirectional  AND SUBCLASSES;
        WITH ATTRIBUTE performanceMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 16 };
```

7.8.17 bidirectionalPerformanceMonitor-vpTTPBidirectional

```
bidirectionalPerformanceMonitor-vpTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASSbidirectionalPerformanceMonitor;
        NAMED BY SUPERIOR OBJECT CLASS vpTTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE performanceMonitorId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 17 };
```

7.8.18 cellLevelProtocolCurrentData-uni

```
cellLevelProtocolCurrentData-uni NAME BINDING
        SUBORDINATE OBJECT CLASS cellLevelProtocolCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS uni AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 18 };
```

7.8.19 cellLevelProtocolCurrentData-interNNI

```
cellLevelProtocolCurrentData-interNNI  NAME BINDING
        SUBORDINATE OBJECT CLASS cellLevelProtocolCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS interNNI  AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 19 };
```

7.8.20 cellLevelProtocolCurrentData-intraNNI

```
cellLevelProtocolCurrentData-intraNNI  NAME BINDING
        SUBORDINATE OBJECT CLASS cellLevelProtocolCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS intraNNI AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 20 };
```

7.8.21 interNNI-managedElementR1

```
interNNI-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS interNNI AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1 AND
        SUBCLASSES;
        WITH ATTRIBUTE interNNIId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 21 };
```

7.8.22 intraNNI-managedElementR1

```
intraNNI-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS intraNNI AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1 AND
        SUBCLASSES;
        WITH ATTRIBUTE intraNNIId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 22 };
```

7.8.23 tcAdaptorTTPBidirectional-managedElementR1

```
tcAdaptorTTPBidirectional-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS tcAdaptorTTPBidirectional AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1
        AND SUBCLASSES;
        WITH ATTRIBUTE tcTTPId;
REGISTERED AS { i751NameBinding 23 };
```

7.8.24 tcAdaptorCurrentData-tcAdaptorTTPBidirectional

```
tcAdaptorCurrentData-tcAdaptorTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS tcAdaptorCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS tcAdaptorTTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 24 };
```

7.8.25 uni-managedElementR1

```
uni-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS uni AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1 AND
        SUBCLASSES;
        WITH ATTRIBUTE uniId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 25 };
```

7.8.26 upcNpcCurrentData-vcCTPBidirectional

```
upcNpcCurrentData-vcCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS upcNpcCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vcCTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 26 };
```

7.8.27 upcNpcCurrentData-vpCTPBidirectional

```
upcNpcCurrentData-vpCTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS upcNpcCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vpCTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 27 };
```

7.8.28 vcCTPBidirectional-vpTTPBidirectional

```
vcCTPBidirectional-vpTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS vcCTPBidirectional AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS vpTTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE vcCTPId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 28 };
```

7.8.29 vcTTPBidirectional-managedElementR1

```
vcTTPBidirectional-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS vcTTPBidirectional AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1 AND
        SUBCLASSES;
        WITH ATTRIBUTE vcTTPId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 29 };
```

7.8.30 vpCTPBidirectional-tcAdaptorTTPBidirectional

```
vpCTPBidirectional-tcAdaptorTTPBidirectional NAME BINDING
        SUBORDINATE OBJECT CLASS vpCTPBidirectional AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS tcAdaptorTTPBidirectional AND SUBCLASSES;
        WITH ATTRIBUTE vpCTPId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 30 };
```

7.8.31 vpTTPBidirectional-managedElementR1

```
vpTTPBidirectional-managedElementR1 NAME BINDING
        SUBORDINATE OBJECT CLASS vpTTPBidirectional AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS "ITU-T M.3100":managedElementR1 AND
        SUBCLASSES;
        WITH ATTRIBUTE vpTTPId;
        CREATE
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 31 };
```

7.8.32 vpVcPMCurrentData-bidirectionalPerformanceMonitor

```
vpVcPMCurrentData-bidirectionalPerformanceMonitor NAME BINDING
        SUBORDINATE OBJECT CLASS vpVcPMCurrentData AND SUBCLASSES;
        NAMED BY SUPERIOR OBJECT CLASS bidirectionalPerformanceMonitor AND SUBCLASSES;
        WITH ATTRIBUTE "Rec. X.739:1993":scannerId;
        CREATE
            WITH-REFERENCE-OBJECT,
            WITH-AUTOMATIC-INSTANCE-NAMING;
        DELETE
            DELETES-CONTAINED-OBJECTS;
REGISTERED AS { i751NameBinding 32 };
```

7.9 Supporting productions

```
AtmMIBMod {itu-t(0) recommendation(0) i(9) atmm(751) informationModel(0) asn1Module(2) atm(0)}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

        -- exports everything
        IMPORTS
        Boolean,
        DisconnectInformation,
        DisconnectResult,
        Failed,
        NameType,
        PointerOrNull,
        ProblemCause
        FROM
        ASN1DefinedTypesModule {ccitt recommendation m(13) gnm(3100)
            informationModel(0) asn1Modules(2) asn1DefinedTypesModule(0)}

        DistinguishedName,
        RelativeDistinguishedName
        FROM
        InformationFramework {joint-iso-ccitt ds(5) modules(1) informationFramework(1)}
        EventTypeId,
        ObjectInstance
        FROM
        CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3)}

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

        i751InformationModel  OBJECT IDENTIFIER ::= {itu-t(0) recommendation i atmm(751)
        informationModel(0) }

        i751standardSpecificExtension OBJECT IDENTIFIER ::= {i751InformationModel
        standardSpecificExtension (0)}
        i751ObjectClass OBJECT IDENTIFIER ::= {i751InformationModel managedObjectClass(3)}
        i751Package OBJECT IDENTIFIER ::= {i751InformationModel package(4)}
        i751Parameter OBJECT IDENTIFIER ::= {i751InformationModel parameter(5)}
        i751NameBinding OBJECT IDENTIFIER ::= {i751InformationModel nameBinding(6)}
        i751Attribute OBJECT IDENTIFIER ::= {i751InformationModel attribute(7)}
        i751Action OBJECT IDENTIFIER ::= {i751InformationModel action(9)}
        i751Notification OBJECT IDENTIFIER ::= {i751InformationModel  notification(10)}

            -- default value definitions

        booleanFalseDefault Boolean ::= FALSE

        booleanTrueDefault Boolean ::= TRUE

        defaultMonitoringBlockSize AverageMonitoringBlockSize ::= decimal128
```

```
        integerZero INTEGER ::= 0

            -- initial value definitions

        booleanFalse Boolean ::= FALSE

            -- additional eventTypes

atmEventType OBJECT IDENTIFIER ::= {i751StandardSpecificExtension 0}

cellHeaderAbnormalityEvent EventTypeId ::= globalForm : {atmEventType 0}

            -- additional characteristicInfo

atmCharacteristicInfo OBJECT IDENTIFIER ::= {i751StandardSpecificExtension 1}

vcCI CharacteristicInformation ::=  {atmCharacteristicInfo 0}
vpCI CharacteristicInformation ::=  {atmCharacteristicInfo 1}

            -- additional probableCause

atmProbableCause OBJECT IDENTIFIER ::= {i751StandardSpecificExtension 2}

lossOfCellDelineation ProbableCause ::= globalValue : {atmProbableCause 0}

            -- supporting productions

        AverageMonitoringBlockSize ::= CHOICE {
            unknown NULL,
            blockSize BlockSize}

        BlockSize ::= ENUMERATED {
            decimal128 (0),
            decimal256 (1),
            decimal512 (2),
            decimal1024 (3)}

        CCProblem ::= ENUMERATED {
            noSourceUserFlow        (0),
            noSinkUserFlow          (1),
            sourceAlreadyActive     (2),
            sinkAlreadyActive       (3) }

        CDVTolerance ::= SEQUENCE {
            cellDelayVariationToleranceCLP0plus1 [0] CDVToleranceCoding OPTIONAL,
            cellDelayVariationToleranceCLP0 [1] CDVToleranceCoding OPTIONAL}

        CDVToleranceCoding ::= CHOICE {
            integerCoding INTEGER,
            floatingPointCoding  FloatingPointCoding}

        CellHeaderAbnormalityType ::= ENUMERATED {
            unassignedVpiVciValue (0),
            outOfRangeVpiVciValue (1)}

        ConnectCtpStatus ::= CHOICE {
            ctpConnected   [0]  ObjectInstance,
            ctpFailed      [1]  ProblemCause}

        Connected ::= SEQUENCE {
            fromTp ObjectInstance,
            toTp    ObjectInstance,
            xCon    ObjectInstance}

        ConnectInformation ::= SEQUENCE OF SEQUENCE {
            fromTermination     [0] CtpOrDescriptor,
            toTermination       [1] CtpOrDescriptor,
            administrativeState         [2] AdministrativeState     OPTIONAL}
```

```
ConnectReply ::= SEQUENCE OF CHOICE {
        connected       [0]  Connected,
        failed     [1]      Failed} -- import from M.3100

ControlCCInformation ::= SEQUENCE {
        activateSourceCCMechanism [0]     BOOLEAN OPTIONAL,
        activateSinkCCMechanism [1]    BOOLEAN OPTIONAL}

ControlCCResult ::= SEQUENCE {
        sourceCCMechanismActive [0]     BOOLEAN     OPTIONAL,
        sinkCCMechanismActive [1]    BOOLEAN       OPTIONAL,
        additionalInformation       SET OF CCProblem      OPTIONAL}

ControlPMInformation ::= SEQUENCE {
        controlSourcePMMechanism [0]     SourcePMMechanism     OPTIONAL,
        controlSinkPMMechanism [1]     SinkPMMechanism     OPTIONAL}

ControlPMResult ::= SEQUENCE {
        sourcePMMechanismResult [0]       SourcePMMechanism       OPTIONAL,
        sinkPMMechanismResult [1]       SinkPMMechanism       OPTIONAL,
        additionalInformation      SET OF PMProblem      OPTIONAL}

CtpOrDescriptor ::=CHOICE {
        ctp [0] ObjectInstance,
        ctpDescriptor [1] Descriptor}

CtpStatus ::= CHOICE {
        disconnected     [0]  NULL,
        failed     [1]  ProblemCause}

Descriptor ::= SEQUENCE {
        interfaceId [0] ObjectInstance, -- a uni, intraNNI, or interNNI
        vpi [1] VpiValue OPTIONAL, -- assigned by managed system if absent
        vci [2]  VciValue OPTIONAL, -- set to 0 for VP cross connect
        egressCDVTolerancePCR [3] CDVTolerance OPTIONAL,
        ingressCDVTolerancePCR [4] CDVTolerance OPTIONAL,
        egressCDVToleranceSCR [5] CDVTolerance OPTIONAL,
        ingressCDVToleranceSCR [6] CDVTolerance OPTIONAL,
        egressMaxBurstSize [7] MaxBurstSize OPTIONAL,
        ingressMaxBurstSize [8] MaxBurstSize OPTIONAL,
        egressPeakCellRate [9] PeakCellRate OPTIONAL,
        ingressPeakCellRate [10] PeakCellRate OPTIONAL,
        egressSustainableCellRate [11] SustainableCellRate OPTIONAL,
        ingressSustainableCellRate [12] SustainableCellRate OPTIONAL,
        egressQosClass [13] QosClass OPTIONAL,
        ingressQosClass [14] QosClass OPTIONAL,
        oamIngressPeakCellRate [15] PeakCellRate OPTIONAL,
        oamEgressPeakCellRate [16] PeakCellRate OPTIONAL,
        oamIngressCDVTolerance [17] CDVTolerance OPTIONAL,
        oamEgressCDVTolerance [18] CDVTolerance OPTIONAL,
        segmentEndPoint [19] Boolean}

DisconnectCtpStatus ::= SEQUENCE OF SEQUENCE {
        ctpInstance       ObjectInstance,
        ctpStatus       CtpStatus}

ExistingCTPs ::= SEQUENCE OF ObjectInstance

FlowDirection::= ENUMERATED {
        outOfSwitch (0),
        inToSwitch (1) }

FloatingPointCoding ::= SEQUENCE {
        e  INTEGER (0..31),
        w INTEGER (0..31)}
```

```
Integer ::= INTEGER

InterfacePointer ::= ObjectInstance -- uni, interNNI, or intraNNI

LoopbackLocation ::= SEQUENCE {
        endPoint BOOLEAN, -- default is TRUE
        loopbackLocationCode OctetStringOrNull -- default is NULL
}

LoopbackOAMCellInfo ::= SEQUENCE {
        loopbackLocation LoopbackLocation,
        oamCellType  OamCellType}

LoopbackOAMCellReply ::= SEQUENCE {
        loopbackSuccessful BOOLEAN,
        problemCause ProblemCause OPTIONAL}

MaxBurstSize ::= SEQUENCE {
        maxBurstSizeCLP0plus1 [0] INTEGER OPTIONAL,
                maxBurstSizeCLP0 [1] INTEGER OPTIONAL}

Null ::= NULL

OamCellType ::= ENUMERATED {
        segment (0),
        endToEnd (1)}

OctetString ::= OCTET STRING

OctetStringOrNull ::= CHOICE {
        octetString OctetString,
        null NULL}

PeakCellRate ::= SEQUENCE {
        peakCellRateCLP0plus1 [0] INTEGER OPTIONAL,
        peakCellRateCLP0 [1] INTEGER OPTIONAL}

PMProblem ::= ENUMERATED {
        noSourceUserFlow          (0),
        noSinkUserFlow        (1),
        sourceAlreadyActive       (2),
        sinkAlreadyActive         (3),
        fePMdataNotAvailable          (4),
        fePMDataProcessingNotSupported        (5),
        backwardReportingAlreadyActive        (6),
        fePMDataProcessingAlreadyActive       (7)}

QosClass ::= ENUMERATED {
        class0 (0),
        class1 (1),
        class2 (2),
        class3 (3),
        class4 (4)}

SourcePMMechanism ::= SEQUENCE {
        sourcePMMechanismStatus       SourcePMMechanismStatus,
        sourceAverageMonitoringBlockSize      AverageMonitoringBlockSize      OPTIONAL}

SourcePMMechanismStatus ::= ENUMERATED {
        deactivate                (0),
        generate              (1),
        generateAndProcessFePMData      (2)}
```

```
SinkPMMechanism ::= SEQUENCE {
        sinkPMMechanismStatus          SinkPMMechanismStatus,
        sinkAverageMonitoringBlockSize      AverageMonitoringBlockSize     OPTIONAL}

SinkPMMechanismStatus ::= ENUMERATED {
        deactivate              (0),
        process             (1),
        processAndReportBackward    (2)}

SustainableCellRate ::= SEQUENCE {
        sustainableCellRateCLP0plus1 [0] INTEGER OPTIONAL,
        sustainableCellRateCLP0 [1] INTEGER OPTIONAL}

VciValue ::= INTEGER (0..65535)

VpiValue ::= INTEGER (0..4095)

END
```

Appendix I

Segment monitoring objects

(This appendix does not form au integral part of this Recommendation)

Segment monitoring is controlled by monitor objects: performance monitoring is controlled by performanceMonitor objects and continuity monitoring is controlled by continuityMonitor objects. Segments are delimited by monitor objects, and the management aplication can control the extension of a segment within an ATM NE where the segment is terminated, thanks to the flowDirection attribute:

– the segment can be terminated as soon as it enters the atmNE, by instantiating the monitor object after the corresponding edge CTP and setting the flowDirection attribute to the "outOfSwitch" value. See Figure I.1:

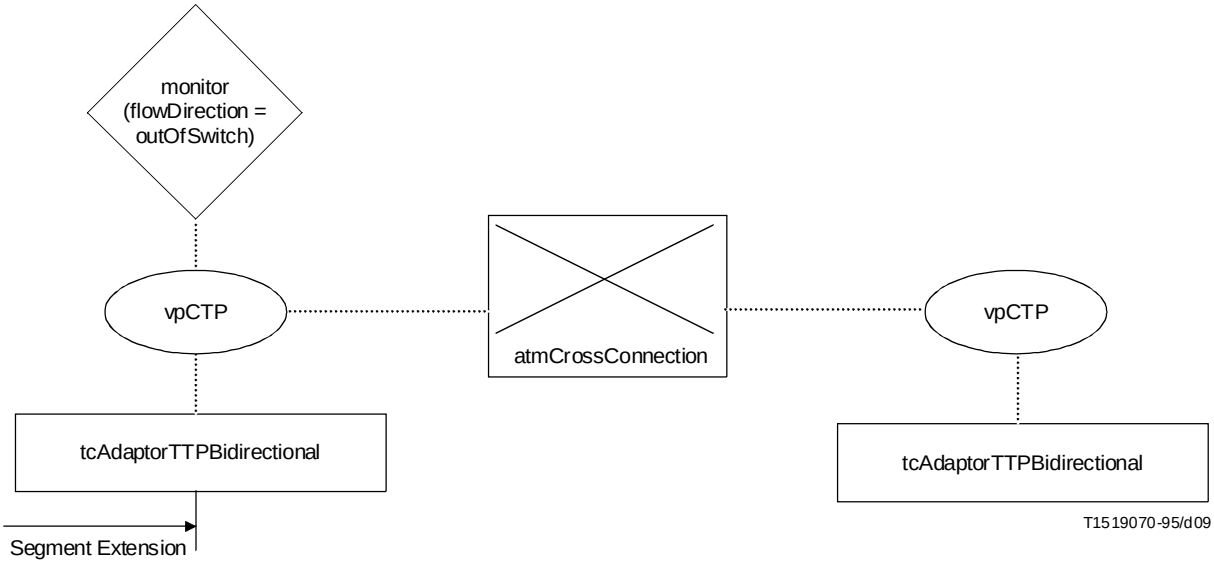


FIGURE I.1/I.751

Case A, the monitor object and the segment extension definition

– the segment can cover the internal cross connection within an ATMNE, by instantiating the monitor object after the cross connected CTP, and setting the flow direction attribute to the "inToSwitch" value. See Figure I.2:

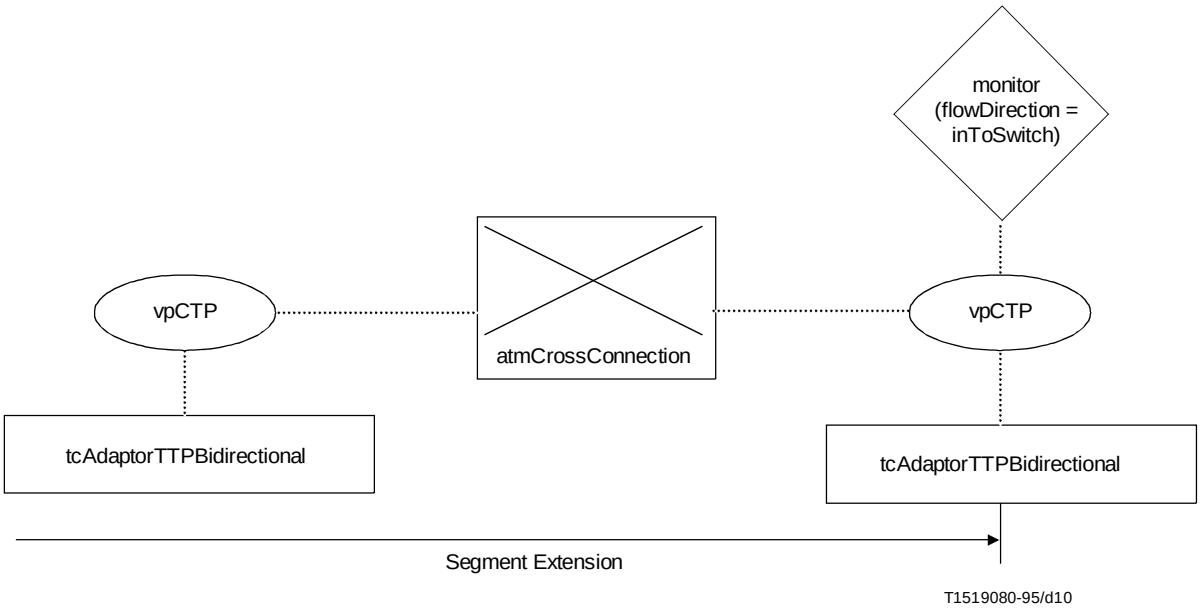


FIGURE I.2/I.751

Case B, the monitor object and the segment extension definition

– if two segments need be monitored and are terminated within a same ATMNE, they can be terminated as shown in Figures I.3 and I.4:

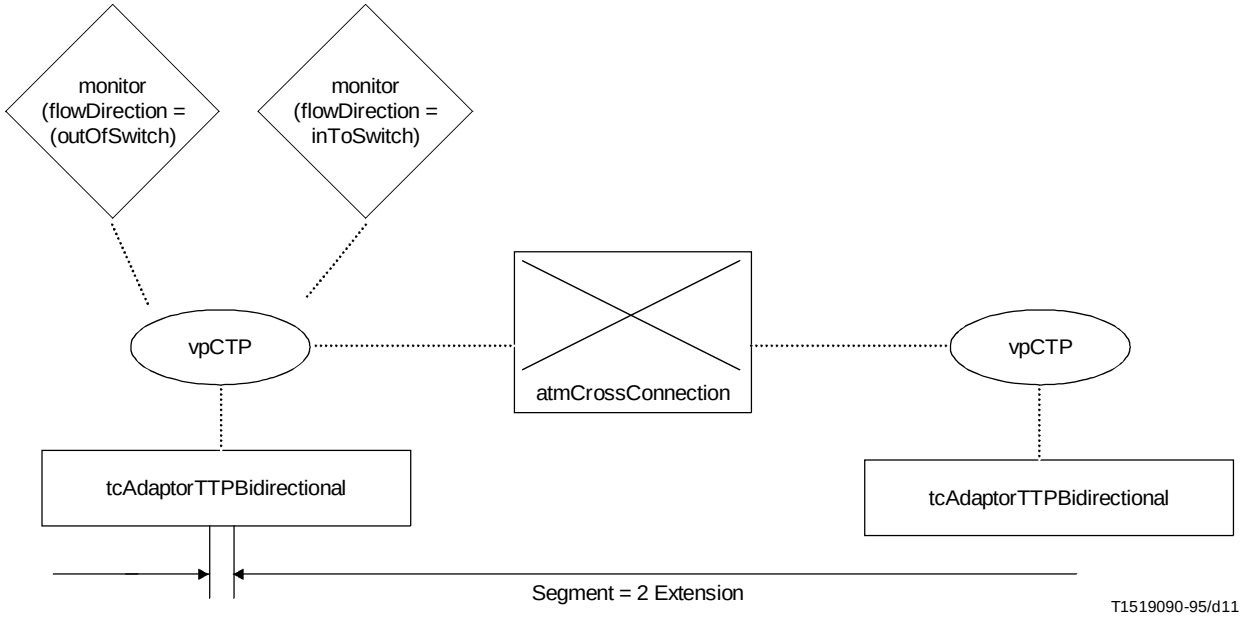


FIGURE I.3/I.751

Case C, the monitor objects and the segment extension definition

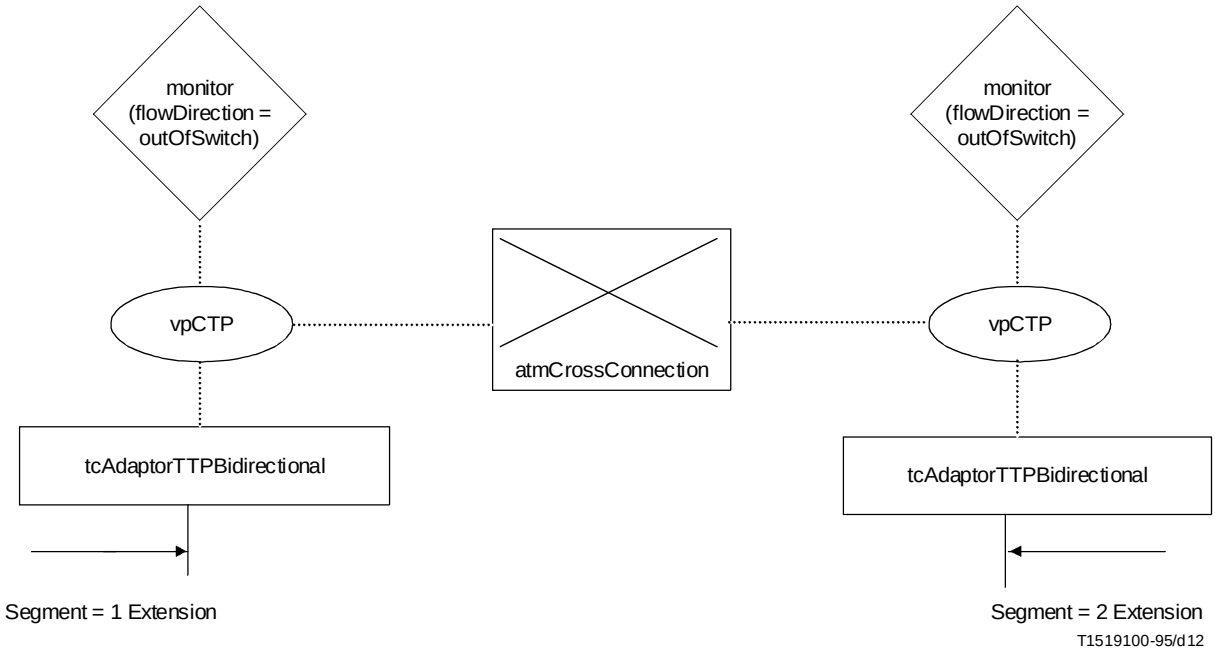


FIGURE I.4/I.751

Case D, the monitor objects and the segment extension definition

– if the segment to be monitored is wholly contained within an ATMNE, monitor objects should be instantiated as depicted in Figure I.5:

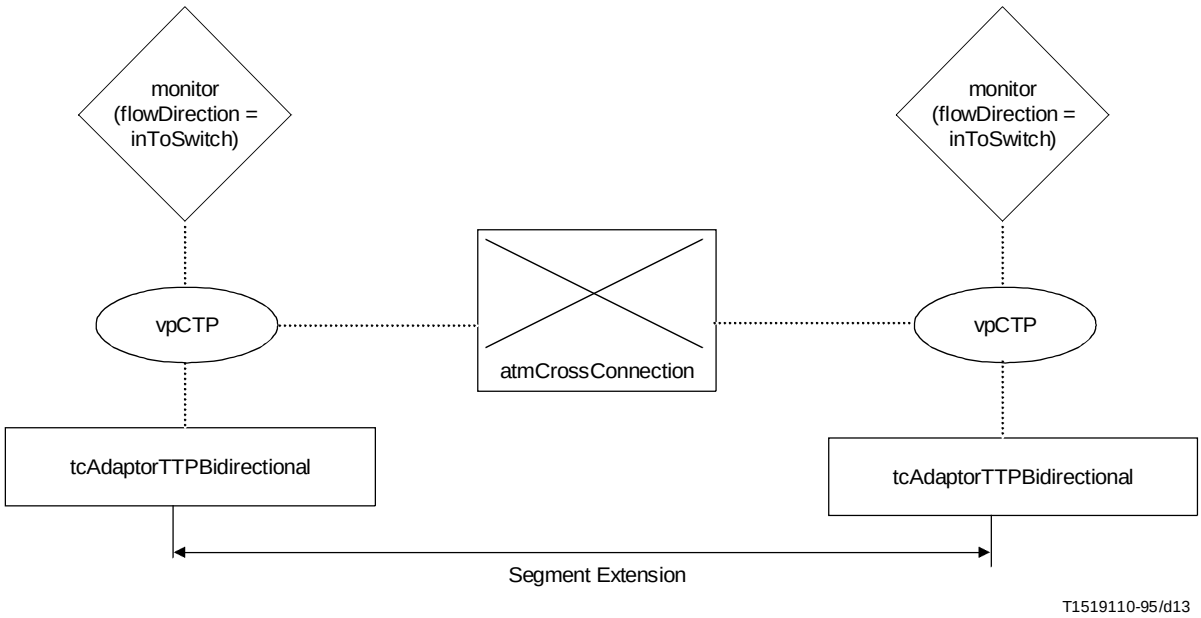


FIGURE I.5/I.751

Case E, the monitor objects and the segment extension definition

Appendix II

Examples of relationship class and relationship mappings

(This appendix does not form au integral part of this Recommendation)

II.1 Characterises relationship

```
characterises       RELATIONSHIP CLASS
BEHAVIOUR     characterisesBehaviour;
ROLE     server
          REQUIRED-ROLE CARDINALITY-CONSTRAINT I751.Cardinality1

ROLE     profile
          REQUIRED-ROLE CARDINALITY-CONSTRAINT I751.Cardinality1
               PERMITTED-RELATIONSHIP-CARDINALITY-CONSTRAINT I751.Cardinality1
REGISTRED AS {IinformationRelationshipClass 01};

characterisesBehaviour       BEHAVIOUR
DEFINED AS "This relationship class is used to characterise  an ATM interface.";

Relationship representation by naming
characterisesMapping         RELATIONSHIP MAPPING
RELATIONSHIP CLASS    characterises;
BEHAVIOUR     characterisesMappingBehaviour;
ROLE profile      RELATED-CLASSES atmAccessProfile
          REPRESENTED-BY NAMING
atmAccessProfile-trailTerminationPointBidirectionalName
          RELATED-TO server,
ROLE server   RELATED-CLASSES trailTerminationPointBidirectional
REGISTERED AS {i751InformationRelationshipMapping 01};

characterisesMappingBehaviour       BEHAVIOUR
DEFINED AS "This relationship mapping indicates how an atmAccessProfile and a characterised Trail
     Termination Point are involved to configure an ATM interface. This Characterised TTP may
     characterise a UNI or an NNI . More than one characterised TTP may be utilised for a given physical
     interface. The ATM Access Profile may not exist independently of the characterised TTP.";

associates          RELATIONSHIP CLASS
BEHAVIOUR     associates Behaviour;
ROLE     server
          REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality1
ROLE     interfaceDescriptor
          REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality1
REGISTRED AS {i751informationRelationshipClass 02};

associates Behaviour    BEHAVIOUR
DEFINED AS "This relationship class associates an interface descriptor e.g. UNI or NNI with a
     configured TTP e.g. tcAdaptor TTP. The interface descriptor and the configured TTP may exist
     independently.";

Relationship representation by a single pointer

tcAdaptorAssociateMapping            RELATIONSHIP MAPPING
RELATIONSHIP CLASS    associates;
BEHAVIOUR       simpleassociatesMappingBehaviour;
ROLE server       RELATED-CLASSES tcAdaptorTTP,

ROLE  interfaceDescriptor  RELATED CLASSES uni, nniInter, nniIntra
     REPRESENTED BY ATTRIBUTE underlyingTTPPointer;
REGISTERED AS {i751InformationRelationshipMapping 02};
```

simpleAssociatesMapping BEHAVIOUR
DEFINED AS "This relationship mapping indicates how a UNI or NNI may set up an ATM interface by association with a TC Adaptor TTP.";

vpAssociatesMapping RELATIONSHIP MAPPING
RELATIONSHIP CLASS associates;
BEHAVIOUR complexAssociatesMappingBehaviour;
ROLE server RELATED-CLASSES vpTTPBidirectional,

ROLE interface descriptor RELATED-CLASSES uni
REPRESENTED-BY ATTRIBUTE underlyingTTPPointer;
REGISTERED AS {i751InformationRelationshipMapping 03};

complexAssociatesMapping BEHAVIOUR
DEFINED AS "This relationship mapping indicates how a UNI may set up a VPTTP Bidirectional as part of an ATM interface where the underlying UNI or NNI is set up independently by a separate association with a TC Adaptor TTP.";

II.2 Client to server relationship

client-server RELATIONSHIP CLASS
BEHAVIOUR client-serverBehaviour;
ROLE client
REQUIRED-ROLE-CARDINALITY CONSTRAINT I751.Cardinality0N
ROLE server
REQUIRED-ROLE-CARDINALITY CONSTRAINT I751.Cardinality1
REGISTRED AS {i751InformationRelationshipClass 03};

client-serverBehaviour BEHAVIOUR
DEFINED AS "This relationship class is used to express the client to layer relationship of transmission entities of different layers as described in ITU-T Recommendation G.805.";

Relationship representation by naming

client-serverMapping RELATIONSHIP MAPPING
RELATIONSHIP CLASS client-server;
BEHAVIOUR client-serverMappingBehaviour;
ROLE client RELATED CLASSES connectionTerminationPoint
REPRESENTED-BY-NAMING client-serverName RELATED-TO server,
ROLE server RELATED-CLASSES trailTerminationPoint;
REGISTERED AS {i751informationRelationshipMapping 04};

II.3 Sublayer client to server relationship

sublayerClient-server RELATIONSHIP CLASS
BEHAVIOUR sublayerClient-serverBehaviour;
ROLE sublayerTTP
REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality01;
ROLE server
REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality11;
REGISTRED AS {i751informationRelationshipClass 04};

sublayerClient-serverBehaviour BEHAVIOUR
DEFINED AS "This relationship class is used to express the client to server relationship between a transmission entity in a server layer and a transmission entity in a client sublayer as described in ITU-T Recommendation G.805.";

Relationship representation by single pointer

sublayerClient-serverMapping RELATIONSHIP MAPPING
RELATIONSHIP CLASS sublayerClient-server;
BEHAVIOUR sublayerClient-serverMappingBehaviour;
ROLE sublayerTTP RELATED CLASSES tcAdaptorTTP,

ROLE server RELATED CLASSES trailTerminationPoint Bidirectional
REPRESENTED BY ATTRIBUTE SupportedByObjectList;
REGISTERED AS {i751informationRelationshipMapping 05};

sublayerClient-serverMapping BEHAVIOUR
DEFINED AS "This relationship mapping indicates how a VC4 TTP can support a tcAdaptor TTP representing the Transmission Convergence sublayer. UNI or NNI may set up an ATM interface by association with a TC Adaptor TTP.";

II.4 Connection relationship

connection RELATIONSHIP CLASS
BEHAVIOUR connectionBehaviour;
ROLE a
REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality1;
ROLE z
REQUIRED-ROLE-CARDINALITY-CONSTRAINT I751.Cardinality1;
REGISTRED AS {i751InformationRelationshipClass 05};

connectionBehaviour BEHAVIOUR
DEFINED AS "This relationship class is used to express the connectivity between termination points in the same G.805 layer.";

Relationship representation by conjugate pointers

TTPtoCTPconnectionMapping RELATIONSHIP MAPPING
RELATIONSHIP CLASS connection ;
BEHAVIOUR TTPtoCTPconnectionMappingBehaviour;
ROLE a RELATED CLASSES trailTerminationPoint
REPRESENTED-BY ATTRIBUTE downstreamConnectivityPointer,
ROLE z RELATED CLASSES connectionTerminationPoint
REPRESENTED-BY ATTRIBUTE upstreamConnectivityPointer;
REGISTERED AS {i751InformationRelationshipMapping 06};

TTPtoCTPConnectionMapping BEHAVIOUR
DEFINED AS "This relationship mapping indicates how a trailTerminationPoint and a connectionTerminationPoint are involed in a connectivity relationship in the same layer.";

CTPtoTTPconnectionMapping RELATIONSHIP MAPPING
RELATIONSHIP CLASS connection;
BEHAVIOUR CTPtoTTPconnectionMappingBehaviour;
ROLE a RELATED CLASSES connectionTerminationPoint
REPRESENTED-BY ATTRIBUTE upstreamConnectivityPointer,
ROLE z RELATED CLASSES trailTerminationPoint
REPRESENTED-BY ATTRIBUTE downstreamConnectivityPointer;
REGISTERED AS {i751InformationRelationshipMapping 07};

CTPtoTTPConnectionMapping BEHAVIOUR
DEFINED AS "This relationship mapping indicates how a connectionTerminationPoint and a trailTerminationPoint are involved in a connectivity relationship in the same layer.";

II.5 Supporting productions

Cardinality1 ::= INTEGER (1)
Cardinality01::=INTEGER(0..1)
Cardinality0N::=INTEGER(0..N)

i751RelationshipClass OBJECT IDENTIFIER ::= {i751InformationModel relationshipClass(11)}
i751RelationshipMapping OBJECT IDENTIFIER ::= {i751InformationModel relationshipMapping(12)}

Appendix III

Multipoint requirements and object classes

(This appendix does not form au integral part of this Recommendation)

The goal of this appendix is to show how the atmFabric object can evolve to handle full multipoint connections.

III.1 Requirements

The operations interface shall support management application requests to establish multipoint VPL and VCL cross-connections in the ATM NE. Provided with each multipoint cross-connect request shall be the following information:

- Multipoint Connection Type (multicast, merge, multicast/merge, full multipoint).
- Primary VPL or VCL Termination Point.

  For broadcast, merge, and multicast/merge cross-connect types, this parameter identifies the VPL or VCL termination point that generates traffic to be broadcasted and/or receives traffic that has been merged. This parameter shall be set to NULL for full multipoint cross-connection types.
- Traffic Descriptors and QOS Class of the Primary Termination Point (if one exists).
- Common VPL or VCL Termination Points.

  This parameter identifies all VPL or VCL termination points involved in the multipoint cross-connection except the primary VPL or VCL termination point.
- Traffic Descriptors and QOS Class for each Common Termination Point.

The operations interface shall support management application requests to tear-down multipoint VPL and VCL cross-connections in an ATM NE.

The operations interface shall support management application requests to individually inhibit and allow the flow of ATM cells to and from each VPL or VCL termination of a multipoint cross-connection.

The operations interface shall support management application requests to add VPL or VCL termination points to an existing multipoint cross-connect arrangement.

The operations interface shall support management application requests to remove VPL or VCL termination points from an existing multipoint cross-connection.

The operations interface should support management application requests to configure specific ATM VP/VC cross-connect relationships as either "recoverable" or "non-recoverable". Recoverable cross-connect relationships remain intact regardless of the operational state of the supporting VP/VC. A non-recoverable cross-connection is one that is automatically torn down (i.e. released) by the ATM NE upon detection of a service affecting failure.

III.2 Managed object class definitions

```
atmFabric MANAGED OBJECT CLASS
DERIVED FROM "Rec. X.721| ISO/IEC 10165-2":top;
CHARACTERIZED BY
        atmFbricPackage  PACKAGE
        BEHAVIOUR atmFabricBeh;
        ATTRIBUTES
            atmFabricId
                GET,
                "Rec. X.721|ISO/IEC-10165-2":administrativeState
                GET-REPLACE,
                "Rec. X.721|ISO/IEC-10165-2":operationalState
                GET,
                "Rec. X.721|ISO/IEC-10165-2":availabilityStatus
                GET;
```

ACTIONS
connect,
disconnect,
connectMultipointBridge,
disconnentMultipointBridge,
addTpsToMultipointBridge,
removeTpsFromMultipointBridge;
;;
REGISTERED AS {};

atmFabric BEHAVIOR
DEFINED AS

"This object class represents the function of managing the establishment and release of ATM cross-connections.

Administrative State:

– Unlocked: The atmFabric is allowed to perform its normal functions. ACTIONS will be accepted to set up, rearrange or remove cross-connections (including multipoint).

– Locked: The atmFabric is not allowed to perform its normal functions. No ACTIONS will be accepted. No new multipoint cross-connections can be set up or removed and no multipoint connections may be rearranged.

Operational State:

– Enabled: When the Fabric is in the enabled operational state, it may be fully operational or partially operational (partially operational is indicated by the availability status attribute).

– Disabled: The Fabric is incapable of performing its normal function. For instance, the managing system will not be able to set up or remove any cross-connection.

Availability Status:

The supported values for this attribute are:

– Degraded: The Fabric is degraded in some respect. For instance, the Fabric cannot perform the function of establishing new cross-connections while it can still accept – ACTIONS to rearrange existing connections. The Fabric remains available for service (i.e. its operational state is enabled) while it is degraded.

– Empty SET (none of the availableStatus conditions exist)."

III.3 Actions

connect ACTION
BEHAVIOR: connectBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX atmMIBMod.ConnectInfo;
WITH REPLY SYNTAX atmMIBMod.ConnectReply;
REGISTERED AS {};

connectBeh BEHAVIOUR
DEFINED AS
"This action is used to establish a point-to-point connection between termination points. The termination points to be connected can be identified explicitly by specifying the associated vcCTPBidirectional object or vpCTPBidirectional object, or by specifying the charcteristics of the termination end points. Multiple point-to-point connections may be requested with a single connect ACTION.

**The two end points can be both explicitly identified
or both be identified with end point descriptors, or a combination of the two.
If an end point descriptor is provided, the NE would reserve necessary
resources, such as VPI and VCI, and automatically create the necessary
vpCTPBidirectional and/or vcCTPBidirectional objects for the the cross
connection.**

**The result, if successful, always returns an explicit list of termination
points.**

**Successful execution of this action would create an instance of the
atmCrossConnection object. This cross-connection object has
the fromTermination and toTermination attributes pointing to the two
termination points. The administrativeState and recoveryType attributes in the
cross-connection object are initialized according to the values provided in
the action request information. If the administrativeState parameter is
omitted, the administrative state will be set to "unlocked".**

**If the administrativeState in the cross-connection object is unlocked, the
upstreamConnectivityPointer and downstreamConnectivityPointer in the two
termination points are set to the local distinguished name of the (peer)
termination point to which it is connected. Also, the
crossConnectionObjectPointer in the termination points points to the
cross-connection object.**

**This action will fail if any of the termination points specified is already
involved in a cross-connection, any of the traffic descriptors specified
cannot be satisfied, or the two termination points do not have compatible
traffic descriptors.
";**

```
disconnect ACTION
        BEHAVIOUR disconnectBeh;
        MODE    CONFIRMED;
        WITH INFORMATION SYNTAX atmMIBMod.DisconnectInfo;
        WITH REPLY SYNTAX atmMIBMod.DisconnectReply;
REGISTERED AS {};

disconnectBeh BEHAVIOUR
        DEFINED AS
"This action is used to take down a point-to-point cross-connection. The
connection to be taken down is specified by identifying a termination point of
the connection. The other termination point of the point-to-point connection
is implicitly disconnected as well and the cross-connection object is deleted.
The connectivity pointers in the disconnected termination points will be set
to NULL as a result of this action. Disconnection of multiple point-to-point
connections can be requested by providing multiple CTP object instances in the
DisconnectInformation. Each component in the DisconnectResult sequence
provides the disconnection result for the corresponding components of the
DisconnectInformation sequence.
This action would not delete any of the termination point objects despite
either termination points may have been created as a result of a previous
connect action.";

connectMultipointBridge ACTION
        BEHAVIOUR connectMultipointBridgeBeh;
        MODE   CONFIRMED;
        WITH INFORMATION SYNTAX atmMIBMod.ConnectMultipointBridgeInfo;
        WITH REPLY SYNTAX atmMIBMod.ConnectMultipointBridgeReply;
REGISTERED AS {};

connectMultipointBridgeBeh BEHAVIOUR
        DEFINED AS
```

"This action is used to establish a multipoint connection between vpCTPBidirectional or vcCTPBidirectional objects. Four types of multipoint connections can be established using this action: broadcast, merge, broadcast/merge (composite), and full multipoint.

The multipointConnectionType attribute of this action is used to identify the desired connection type. Note that this action will result in the cross-connection of CTP objects to an instance of the multipointBridge object.

Primary CTP – This parameter identifies a termination point (vpCTPBidirectional or vcCTPBidirectional object) or an end point descriptor for the termination point that generates traffic to broadcase and/or receives merged traffic for broadcast, merge, or composite multipoint connection types. For full multipoint connections (i.e. all legs communicate with all other legs), the value of this attribute shall be set to NULL. The termination point identified derived from this parameter is contained in the primaryCTP attribute of the multipointBridge object.

Common CTPs – This identifies end point descriptor or CTP object instances (vpCTPBidirectional object class or vcCTPBidirectional object class) of all legs of the multipoint connection except the leg identified via the primaryCTP attribute. For full multipoint cross-connections, all the legs of the multipoint cross-connection shall be identified by this attribute. The termination points derived from this parameter are contained in the commonCTPs attributes of the multipointBridge.

Administrative State – This parameter will be used as the initial value for the administrativeState attributes in multipointBridge and its associated cross-connection objects.

The action will fail if the primaryCTP is specified but cannot be connected or none of the commonCTPs can be connected. If the action is accepted, the result would return the primaryCTP termination point, the connected common CTPs, and a problem cause for each of the non-connected common CTPs.";

```
disconnectMultipointBridge ACTION
        BEHAVIOUR disconnectMultipointBridgeBeh;
        MODE CONFIRMED;
        WITH INFORMATION SYNTAX atmMIBMod.DisconnectMultipointBridgeInfo;
        WITH REPLY SYNTAX atmMIBMod.DisconnectMultipointBridgeReply;
        REGISTERED AS {};

disconnectMultipointBridgeBeh BEHAVIOUR
        DEFINED AS
"This action is used to release a multipoint cross-connection. Supplied
with this action is the instance of the multipointBridge object supporting
the multipoint cross-connection.

If any of the commonCTPs cannot be disconnected, the primaryCTP, if any,
should not be disconnected. If the multipointBridge is not completely
disconnected, the action result would indicate which commonCTPs have been
disconnected and which commonCTPs are not disconnected and why.";

addTpsTomMultipointBridge ACTION
        BEHAVIOUR addTpsToMultipointBridgeBeh;
        MODE CONFIRMED;
        WITH INFORMATION SYNTAX atmMIBMod.addTpsToMultipointBridgeInfo;
        WITH REPLY SYNTAX atmMIBMod.addTpsToMultipointBridgeReply;
        REGISTERED AS {};

addTpsToMultipointBridgeBeh BEHAVIOUR
        DEFINED AS
"This action is used to add one or more CTP to the identified
multipoint connection.

Supplied with this action is the following information:
```

New CTPs – This parameter identifies the additional CTPs to add to the existing multipoint connection.

MultipointBridge – This parameter identifies the instance of the multipointBridge object class to which the identified legs should be added.

If the request is granted, the commonCTPs attributes, in the multipoint bridge object, shall be reset to reflect the new legs of the multipoint connections.";

removeTpsFromMultipointBridge ACTION
BEHAVIOUR removeTpsFromMultipointBridgeBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX atmMIBMod.RemoveTpsFromMultipointBridgeInfo;
WITH REPLY SYNTAX atmMIBMod.RemoveTpsFromMultipointBridgeReply;
REGISTERED AS {};

RemoveTpsFromMultipointBridgeBeh BEHAVIOUR
DEFINED AS
"This action is used to remove one or more legs from the identified multipoint connection.

Supplied with this action is the following information:

Existing CTPs – This parameter identifies the existing CTPs to remove from the multipoint connection.

MultipointBridge – This parameter identifies the instance of the multipointBridge object class from which the identified legs should be removed (disconnected).

If the request is granted, the commonCTPs attributes, in the multipoint bridge object, shall be reset to reflect the remaining legs of the multipoint connections.";