

ITU-T

G.7718.1/Y.1709.1

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(12/2006)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA,
DIGITAL SYSTEMS AND NETWORKS

Data over Transport – Generic aspects – Transport
network control aspects

SERIES Y: GLOBAL INFORMATION
INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS
AND NEXT-GENERATION NETWORKS

Internet protocol aspects – Operation, administration and
maintenance

Protocol-neutral management information model for the control plane view

ITU-T Recommendation G.7718.1/Y.1709.1

ITU-T G-SERIES RECOMMENDATIONS
TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
General	G.7000–G.7099
Transport network control aspects	G.7700–G.7799
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999

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

ITU-T Recommendation G.7718.1/Y.1709.1

Protocol-neutral management information model for the control plane view

Summary

ITU-T Recommendation G.7718.1/Y.1709.1 provides a protocol-neutral management information model for managing the ASON control plane. It includes an electronic attachment containing the UML description in navigable html format and in the original Rational Rose format.

Source

ITU-T Recommendation G.7718.1/Y.1709.1 was approved on 14 December 2006 by ITU-T Study Group 15 (2005-2008) under the ITU-T Recommendation A.8 procedure.

Keywords

ASON, control plane, protocol-neutral information model, UML.

.....

FOREWORD

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

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

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

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

NOTE

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

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

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

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2007

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

Page

1	Scope	1
2	References.....	1
3	Definitions	2
3.1	Terms defined elsewhere	2
3.2	Terms defined in this Recommendation.....	2
4	Abbreviations.....	2
5	Conventions.....	4
5.1	Object creation and deletion	4
5.2	Textual conventions.....	4
6	Control plane management requirements	8
7	Analysis	8
8	Design of the protocol-neutral information model	9
8.1	Class diagram – Call/connection class diagram	9
8.2	Class diagram – Common attributes diagram (InheritanceTree)	10
8.3	Class diagram – Discovery service class diagram.....	11
8.4	Class diagram – Notification class diagram	11
8.5	Class diagram – Routing service class diagram	12
8.6	Class diagram – Signalling service class diagram.....	13
9	Class definitions.....	13
9.1	AlarmSeverityAssignmentProfile.....	13
9.2	CR-LDPProtocol	13
9.3	Call	14
9.4	CallCreateData	19
9.5	CallDetailRecord	21
9.6	CallService	22
9.7	CommonResourceInformation	26
9.8	Connection.....	27
9.9	ConnectionCreateData.....	29
9.10	ConnectionService.....	31
9.11	ConnectionTerminationPoint	32
9.12	ControlPlane	32
9.13	ControlPlaneService	34
9.14	DiscoveryService.....	35
9.15	ISISArea	36
9.16	ISISNeighbour	38
9.17	ISISProtocol	39
9.18	LayerAdjacencyDiscovery	40
9.19	ManagedElement.....	42



Documents

Document Sharing Hub

ITU-T Rec. G.7718.1/Y.1709.1 (12/2006)

iii

	Page
9.20 OSPFArea.....	42
9.21 OSPFNeighbour	44
9.22 OSPFProtocol	45
9.23 PNNIArea	46
9.24 PNNINeighbour.....	48
9.25 PNNIRoutingProtocol	49
9.26 PNNISignallingProtocol.....	50
9.27 RSVPProtocol.....	51
9.28 RoutingArea	51
9.29 RoutingTable	55
9.30 SNP	55
9.31 SNPLinkConnection.....	56
9.32 SNPP	57
9.33 SNPPLink	58
9.34 SignallingProtocol	62
9.35 TerminationPoint	63
9.36 TrailTerminationPoint	63
9.37 TransportEntityCapabilityExchange	63
10 Notification definitions.....	65
10.1 Alarm.....	65
10.2 AttributeValueChangeNotification.....	66
10.3 Notification.....	66
10.4 ObjectCreationNotification	68
10.5 ObjectDeletionNotification	68
10.6 StateChangeNotification.....	68
11 Data Types.....	68
11.1 Enumerations.....	68
11.2 Structures.....	75
11.3 TypeDefinitions.....	76
Appendix I – G.7718.1 and TMF 608v3.0.....	79
Appendix II – Alternate representations	80

Electronic attachment:

- Layer rate supporting document.
- UML description in navigable html format.
- UML description in original Rational Rose format.

ITU-T Recommendation G.7718.1/Y.1709.1

Protocol-neutral management information model for the control plane view¹

1 Scope

This Recommendation provides a protocol-neutral management information model for managing the ASON control plane. It identifies the telecommunication management network (TMN) managed entities required. These entities are relevant to information exchanged across standardized interfaces defined in [ITU-T M.3010] TMN architecture. The protocol-neutral management information model should be used as the base for defining protocol-specific management information models, for example, CMISE information model and CORBA IDL interfaces.

The specific mapping of the protocol-neutral entities into protocol-specific managed object classes is the decision of the protocol-specific modelling design. Protocol-specific information models and their mapping from the protocol-neutral model will be described in other documents.

This Recommendation applies to network elements performing ASON control plane functions, and those systems in the TMN that manage them. The architecture of ASON control plane and its components are given in [ITU-T G.8080] and the management framework and requirements are given in [ITU-T G.7718].

The object entities defined in this model apply to configuration and fault management.

There are several different perspectives from which management information may be defined for management purposes. The control plane fragment contains those object entities and information having to do with managing the control plane functions, irrespective of the platform performing those functions. That platform may be a traditional network element or a stand-alone element with no transport functionality.

2 References

The following ITU-T 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 references are subject to revision; 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. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [ITU-T G.7710] ITU-T Recommendation G.7710/Y.1701 (2001), *Common equipment management function requirements*.
- [ITU-T G.7713] ITU-T Recommendation G.7713/Y.1704 (2006), *Distributed call and connection management (DCM)*.
- [ITU-T G.7714] ITU-T Recommendation G.7714/Y.1705 (2005), *Generalized automatic discovery for transport entities*.
- [ITU-T G.7718] ITU-T Recommendation G.7718/Y.1709 (2005), *Framework for ASON management*.
- [ITU-T G.8080] ITU-T Recommendation G.8080/Y.1304 (2006), *Architecture for the automatically switched optical network (ASON)*.

¹ This Recommendation includes an electronic attachment containing the layer rate supporting document, the UML description in navigable html format and in the original Rational Rose format.

- [ITU-T M.3010] ITU-T Recommendation M.3010 (2005), *Principles for a telecommunications management network*.
- [ITU-T M.3100] ITU-T Recommendation M.3100 (2005), *Generic network information model*.
- [ITU-T X.733] ITU-T Recommendation X.733 (1992), *Information technology – Open Systems Interconnection – Systems management: Alarm reporting function*.
- [AF-PNNI-0055] MFA Forum AF-PNNI-0055.002, *Private Network-Network Interface Specification Version 1.1 (PNNI 1.1)*, April 2002.
- [IETF RFC 2328] IETF RFC 2328 (1998), *OSPF Version 2*.
- [IETF RFC 3473] IETF RFC 3473 (2003), *Generalized Multi-Protocol Label Switching (GMPLS) Signalling Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions*.
- [ISO/IEC 10589] ISO/IEC 10589:2002, *Information technology – Telecommunications and information exchange between systems – Intermediate system to intermediate system intra-domain routing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473)*.
- [TMF MTNM] Tele Management Forum MTNM SD1-17_LayerRates.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

- 3.1.1 alarm severity assignment profile (ASAP):** [ITU-T M.3100].
- 3.1.2 connection termination point (CTP):** [ITU-T M.3100].
- 3.1.3 termination point (TP):** [ITU-T M.3100].
- 3.1.4 trail termination point (TTP):** [ITU-T M.3100].
- 3.1.5 alarm reporting control (ARC):** [ITU-T G.7710].
- 3.1.6 subnetwork point (SNP):** [ITU-T G.8080].
- 3.1.7 subnetwork point pool (SNPP):** [ITU-T G.8080].
- 3.1.8 subnetwork point pool link (SNPP link):** [ITU-T G.8080].

3.2 Terms defined in this Recommendation

None.

4 Abbreviations

This Recommendation uses the following abbreviations:

ASON	Automatically Switched Optical Network
AVC	Attribute Value Change Notification
AVP	Attribute name Value Pair
CMISE	Common Management Information Service Element
CORBA	Common Object Request Broker Architecture

CP	Control Plane
CR-LDP	Constraint-based Label Distribution Protocol
DCM	Distributed Call and Connection Management
EMS	Element Management System
ENNI	External Network-Network Interface
GDMO	Guidelines for Definition of Managed Objects
HTML	HyperText Markup Language
Id, ID	Identifier
IDL	Interface Definition Language
I-NNI	Internal Network-Network Interface
IP	Internet Protocol
ISIS	Intermediate System – Intermediate System routing protocol
LCAS	Link Capacity Adjustment Scheme
LRM	Link Resource Manager
LSA	Link State Advertisement
MIB	Management Information Base
MTNM	Multi-Technology Network Management
MTU	Maximum Transmission Unit
NA	Not Applicable
NE	Network Element
NMS	Network Management System
OSPF	Open Shortest Path First
PNNI	Private Network-to-Network Interface
RSVP	Resource Reservation Protocol
RSVP-TE	Resource Reservation Protocol – Traffic Engineering
SCN	Signalling Communication Network
SRG	Shared Risk Group
TCE	Transport Capability Exchange
TMF	TeleManagement Forum
TMN	Telecommunication Management Network
TRI	Transport Resource Identifier
TTP	Trail Termination Point
UML	Unified Modelling Language
UNI	User-to-Network Interface
UNI-C	User-to-Network Interface – Customer side
UNI-N	User-to-Network Interface – Network side



5 Conventions

This Recommendation uses the UML notation to specify a protocol-neutral information model. UML itself does not specify an execution environment, so there are no pre-existing object methods provided. This is in contrast to the X.721 (GDMO/CMISE) notation which provides many methods in its assumed environment.

This Recommendation has therefore specified some general methods, and has specified them as operations against the ControlPlane object.

5.1 Object creation and deletion

An object can be Created or Deleted by using the relevant operation on the ControlPlane object.

However, some objects are created as the result of very significant activity in the control plane, and the generic create operation cannot have any knowledge of the very special behaviour of (for example) creating a call object.

Some objects are also expected to be automatically created when the platform supporting those objects is initialized. It is not expected to create such objects by independent management action.

This specification takes care of these significant differences by providing special operations to create these special objects, and it is a convention that these special operations are used in place of the generic ones.

5.2 Textual conventions

Several important properties of network management applications need to be specified, and UML does not provide any formal support. This Recommendation uses some English text conventions to provide for this specification.

In all cases, these specifications are inherited via the normal inheritance relationship, though the UML notation does not provide any formal support for this inheritance.

For the purpose of this Recommendation, these pieces of specification achieved by the English text are referred to as "properties".

5.2.1 Attribute properties

In network management applications, controlling access to attributes is critical; however, UML does not provide any formal control, and so we resort to textual conventions in the object descriptions. The three Booleans Readable, Writable and Invariant appear in the object descriptions as below. The values are "Yes" or "No."

- Readable by EMS? Yes
- Writable by EMS? Yes
- Invariant? No

When Writable = yes and Invariant = yes together, this signifies an attribute that can be written once and that the EMS can provide a one time configuration value.

When Writable = No and Invariant = Yes, this signifies an attribute which receives a value at object creation time, or as a result of a side effect of creation, and that it is not expected to change over time.

When Writable = No and Invariant = No signifies an attribute whose value reflects a changing value in the managed object, and whose value cannot be affected by the EMS.

In network management applications, it is common that an object will autonomously report an attribute value change. Mentioning the attribute value change (AVC) notification in the Notification Type property description specifies this behaviour.

Attributes whose value should be set at object creation to reflect the current value in the system are shown as having a default value of "NA" (not applicable).

Note that the method of ensuring compliance with this access rights specification is left to protocol specific implementations.

5.2.2 Complex attribute types

Complex attributes, which are used in more than one object, are listed in clause 11 as if they were objects. The fields of the complex type become the attributes of the type object.

5.2.3 Object properties

Several properties of objects are interesting to network management applications, and these are added by textual conventions.

Object Lifecycle: indicates when the object is created, and by whom and when the object is deleted.

Notifications specify autonomous behaviour of the object, perhaps in response to a private internal method. In particular, Creation and Deletion are frequently notified to the management system. This behaviour is specified by the Booleans "Object Creation Notification Issued" and "Object Deletion Notification Issued".

Inheritance is indicated by the "Derived from" note.

5.2.4 Operation properties

Operations have several aspects to their specification that are subject to conventions.

Pre-condition

The pre-condition in this Recommendation specifies the conditions that must be true before the operation is invoked. In case that no pre-condition is specified, "TRUE" is assumed. That is, the operation user need not take any particular steps to check any state before the operation is issued.

Post-condition

The post-condition is a condition or predicate that is always true after an operation has been performed successfully.

Exceptions

This describes various reasons that the operations may have failed, and conventionally includes a restatement of the pre-condition.

Idempotence

Operations are all expected to be idempotent, so setting a value will not result in an error if the operation does not actually cause a value change. In computing, idempotence is the quality of something that has the same effect if used multiple times as it does if used only once, similar to the idempotence notion in mathematics.

Exceptions

Operations support the following default exceptions.

notImplemented, invalidInput, and error. These exceptions have their usual meaning and the error exception is unspecified. Operations requiring a more detailed error exception specify this separately. Operations not requiring special exception behaviour have the operation exception description "Standard".

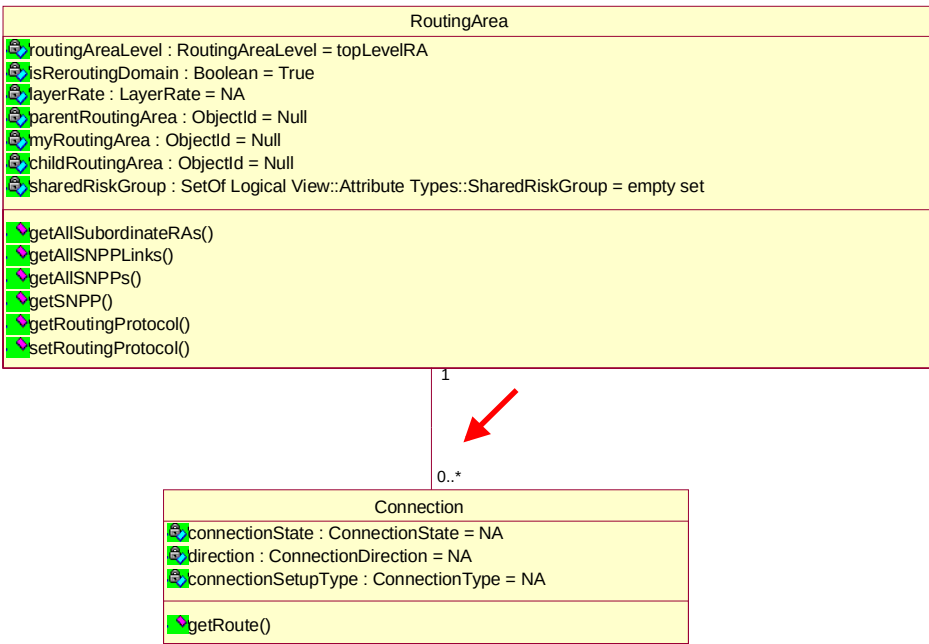
5.2.5 Relationships

Relationships among objects are indicated by UML associations; however, not every association is intended to be navigable in a protocol specific implementation. Navigability requires that the protocol specific design must include attributes (and possible operations) that model the relation. The means by which these specifications are implemented is left to the protocol specific design. Rather than confuse the UML information model with relation navigation details, which are not automatically checked for correctness, this Recommendation uses some conventions to indicate which relationships should be navigable, and which are explanatory.

The conventions are as follows:

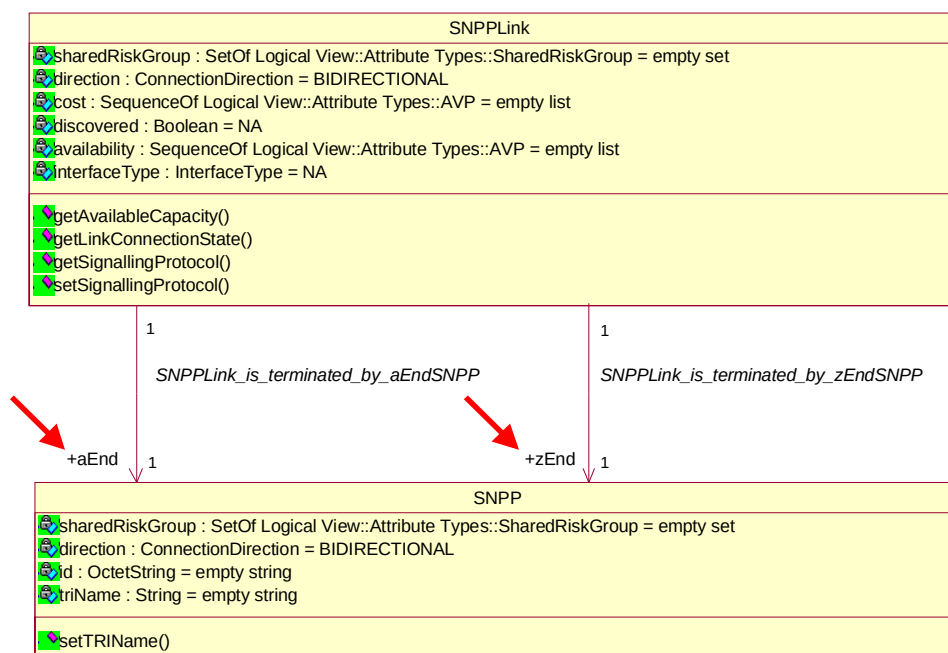
- 1) All Associations and Relations without any role names are just for explanation to the reader of the model and will not be implemented unless a specific operation or attribute requires it.

Example:



- 2) A public role name (+) indicates that the association will be navigable in the protocol specific design. The method of implementing the navigation (attribute or operation) is left to the specific design.

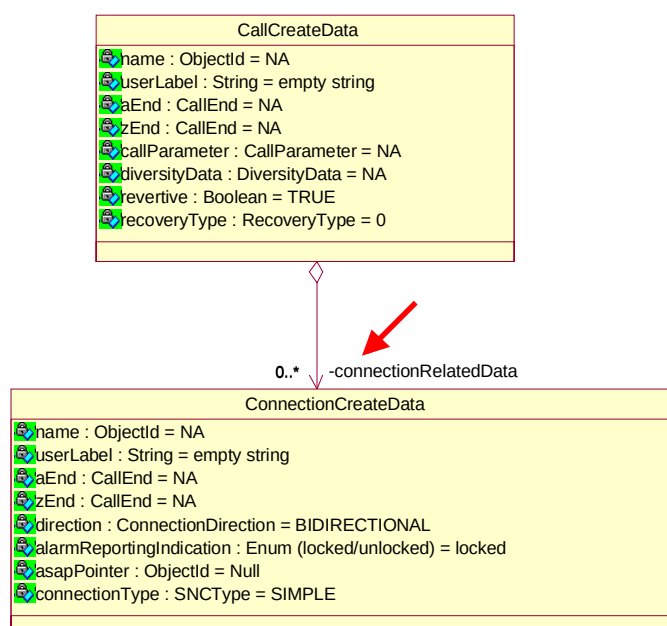
Example:



IDL: aEnd and zEnd will become attributes of the SNPPLink struct containing the name of the a-End and z-End SNPP.

- 3) If class A has a Containment or Composition relationship with class B and class B plays a private (-) role in this relationship, then the contained information of class B shall be part of the information when class A is retrieved via the interface (i.e., attributes of class B shall be retrieved together with attributes of class A).

Example:



6 Control plane management requirements

This Recommendation models the control functions that are relevant. These functions are defined in [ITU-T G.7718]. General details of the management functions that need to be modelled are provided in [ITU-T G.7710].

7 Analysis

In this Recommendation, managed resources and management support resources are modelled as objects in the information model. The management view of a resource is a managed object. This Recommendation specifies the properties of the resources visible for management. Objects with similar properties are grouped into object classes. An object instance is an instantiation of an object class. The properties of an object include the behaviour, attributes, and operations that can be applied on the object. An object instance is characterized by its object class and may possess multiple attribute types and associated values. In the protocol-neutral model, object classes are represented as unified modelling language (UML) classes.

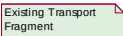
Object classes, attribute types, and operations are defined for the purpose of communicating network management messages between systems, and need not be related to the structure of data within those systems.

An object class may be a subclass of another class. A subclass inherits properties of its superclass, in addition to possessing its own specific attributes and properties. In this Recommendation, the control plane specific object classes are defined. In the future, when defining protocol-specific object classes, the protocol-specific object classes could be mapped from the protocol-neutral object classes and also inherited from the protocol-specific generic transport object classes for additional properties.

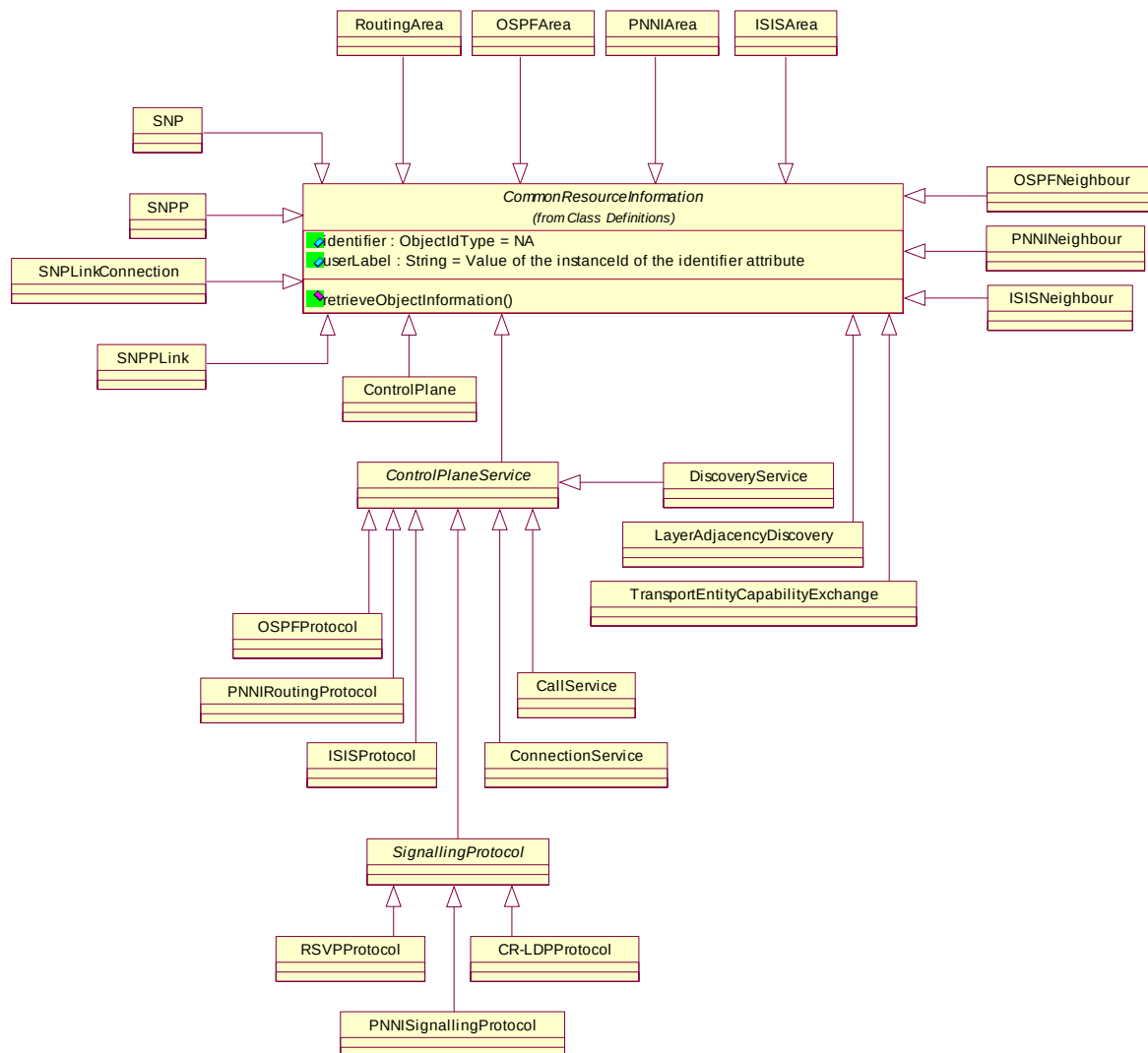
At the protocol specific design stage, designers have the freedom to change the structure of information specified in this Recommendation as long as the specified semantics is maintained. This allows for protocol specific design optimizations to be performed. For example, this specification returns object identifiers, while a possible design optimization could be to return the whole object and so avoid a second transaction.

In addition, to the control plane resources, the model also refers to object classes for management support functions such as alarm reporting control and alarm severity assignment.

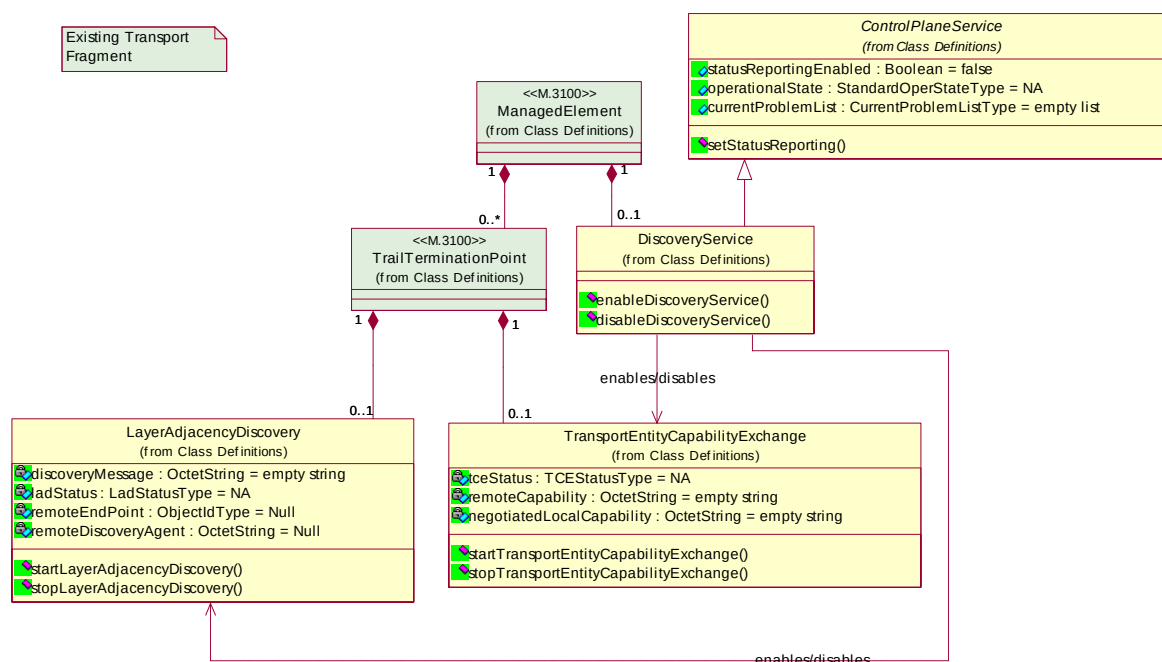
8.1 Class diagram – Call/connection class diagram



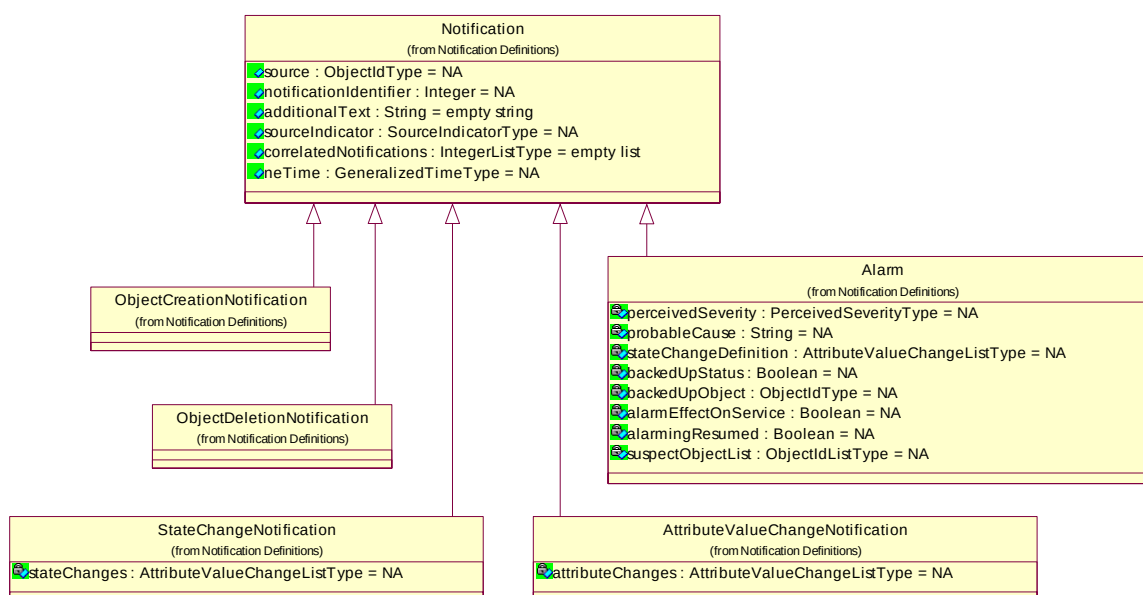
8.2 Class diagram – Common attributes diagram (InheritanceTree)



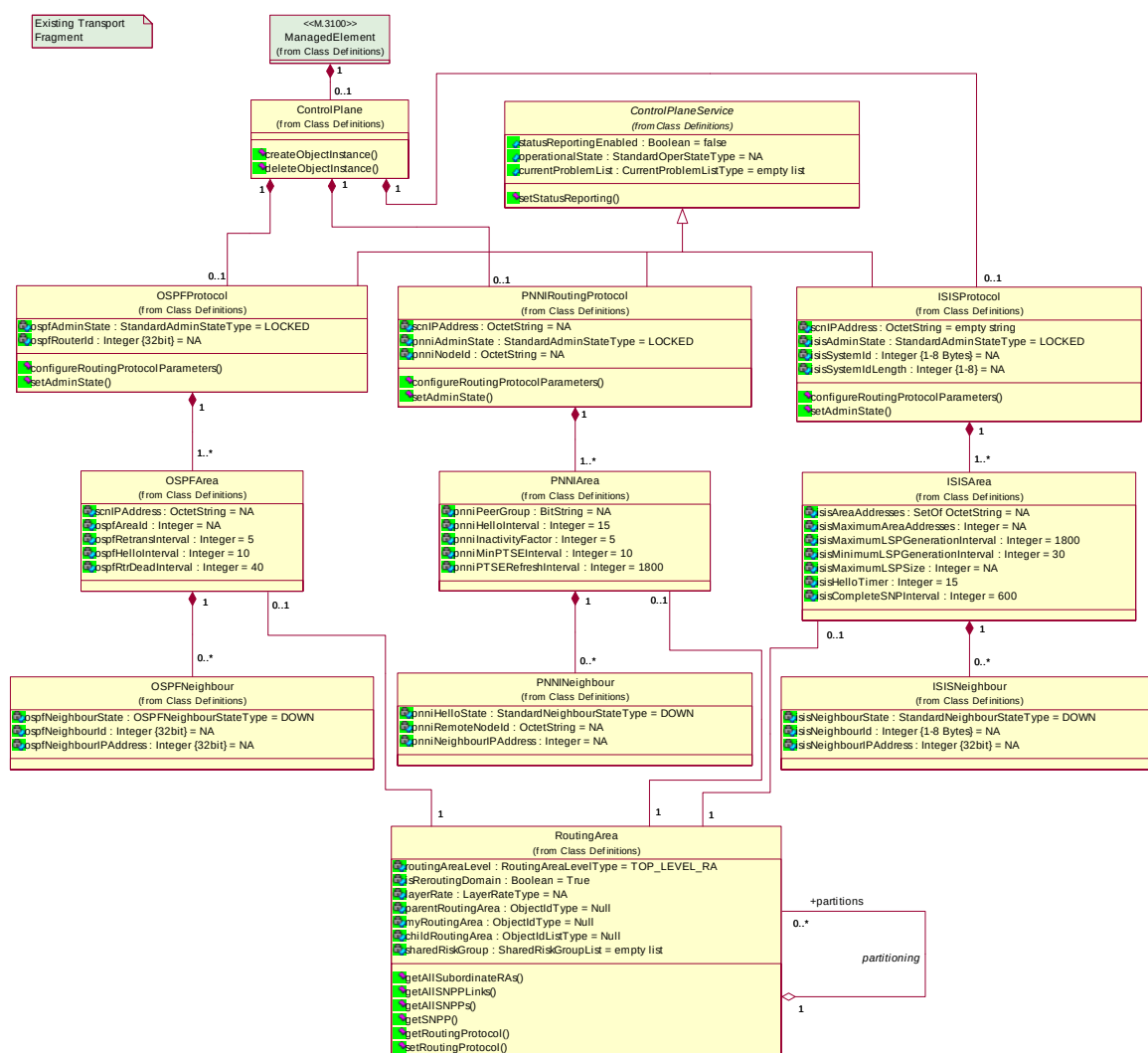
8.3 Class diagram – Discovery service class diagram



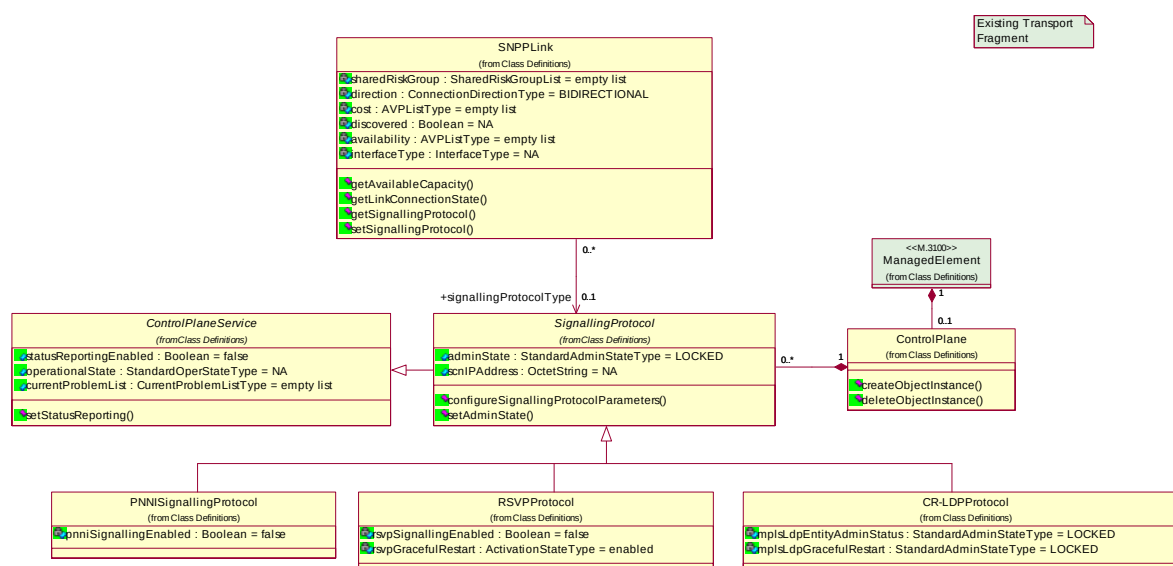
8.4 Class diagram – Notification class diagram



8.5 Class diagram – Routing service class diagram



8.6 Class diagram – Signalling service class diagram



9 Class definitions

9.1 AlarmSeverityAssignmentProfile

The AlarmSeverityAssignmentProfile object specifies the alarm severity assignment for managed objects. Instances of this object are referenced by the asapPointer attribute in the managed objects.

9.1.1 Associations

9.1.1.1 (Unnamed Target Role) [0..*]: Unnamed Association (Unnamed Supplier Role) to Connection

9.1.1.2 (Unnamed Target Role) [0..*]: Unnamed Association (Unnamed Supplier Role) to ConnectionCreateData

9.1.1.3 (Unnamed Target Role) [0..*]: Unnamed Association (Unnamed Supplier Role) to Call

9.1.1.4 (Unnamed Target Role) [0..*]: Unnamed Association (Unnamed Supplier Role) to CallCreateData

9.2 CR-LDPPProtocol

The CR-LDP protocol object provides a management view of the specific attributes of a CR-LDP protocol instance.

Object Lifecycle: Could be created and deleted by Management Action

Object Creation Notification Issued? Yes

Object Deletion Notification Issued? Yes

Derived from: SignallingProtocol

9.2.1 Attributes

9.2.1.1 **mplsLdpEntityAdminStatus: StandardAdminStateType = LOCKED**

If set to 'unlocked', then Entity will attempt to establish a new session with the Peer.

Readable by EMS? Yes

Writeable by EMS? Yes

Invariant? No

Notification Type? None

9.2.1.2 **mplsLdpGracefulRestart: StandardAdminStateType = LOCKED**

Flag to enable/disable the graceful restart procedure.

Readable by EMS? Yes

Writeable by EMS? Yes

Invariant? No

Notification Type? None

9.3 Call

The Call Object represents the service between two end points in the network. Calls are identified globally.

Object Lifecycle: The Call object is the result of an establishCall operation. The Call object exists for the lifetime of the Call.

Object Creation Notification Issued? No; it is expected that Calls are available for inspection when required, but due to potential high call volume, individual calls do not issue notifications.

Object Deletion Notification Issued? No

Derived from: CommonResourceInformation

9.3.1 Attributes

9.3.1.1 **connectionSetupType: ConnectionKindType = NA**

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.2 **callState: CallStateType = NA**

This attribute represents the state of the Call. The Call state depends on the state of the associated Connections.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? No

Notification Type? None

9.3.1.3 **aEnd: CallEndType = NA**

This attribute represents the source end point of the Call. The value of this attribute shall be in one of the following name types:

- SNPP;
- SNP;
- TRI Name, which identifies either a TRI or a Group-TRI Name.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.4 **zEnd: CallEndType = NA**

This attribute represents the sink end point of the Call. The value of this attribute shall be in one of the following name types:

- SNPP;
- SNP;
- TRI Name, which identifies either a TRI or a Group-TRI Name.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.5 **alarmReportingIndication: StandardAdminStateType = LOCKED**

Provides an indication of whether alarm reporting for this Call is administratively activated or de-activated.

Unlocked = alarm reporting is activated; locked = alarm reporting is de-activated.

Readable by EMS? Yes

Writeable by EMS? Yes

Invariant? No

Notification Type? AVC

9.3.1.6 **asapPointer: ObjectIdType = Null**

This attribute indicates the assignment of an Alarm Severity Assignment Profile to the Call.

Readable by EMS? Yes

Writeable by EMS? Yes

Invariant? No

Notification Type? AVC

9.3.1.7 **originator: PlaneType = NA**

This attribute indicates the origin of the Call create request.

NOTE – The attribute has the scope of an individual NE.

Allowed values are ControlPlane or ManagementPlane.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.8 callParameter: CallParameterType = NA

This attribute identifies the requested thresholds (wrt. the number of Connections) of the Call:

- Degraded Threshold;
- Severely Degraded Threshold.

Readable by EMS? Yes

Writeable by EMS? Yes

Invariant? No

Notification Type? AVC

9.3.1.9 diversityData: DiversityDataType = NA

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.10 revertive: Boolean = True

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.1.11 recovery: RecoveryType = NONE

Allowed values are none (0), restorable (1), protected (2).

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.3.2 Associations

9.3.2.1 (Unnamed Target Role) [0..*]: Unnamed Association (Unnamed Supplier Role) to Connection

9.3.2.2 (Unnamed Target Role) [1]: Unnamed Association (Unnamed Supplier Role) to CallService

9.3.3 Operations

9.3.3.1 getConnection (name: ObjectIdType = NA, connection: Connection = Null): void

This operation retrieves the details of a Connection given a specified Connection name.

NOTE – (In a control plane environment) The managed element has to control the originating end of the Call otherwise the Connection information is not available.

Operation exceptions:

Standard

Parameter description:

name

/ in */* The name of the Connection to be retrieved.

connection

/ out */* The Connection object requested.

Pre-conditions:

None

Post-conditions:

The system remains unchanged.

9.3.3.2 addConnections (connectionCreateDataList: SequenceOf ConnectionCreateData = NA, connectionRouteRearrangementAllowed: Boolean = False, createdConnections: SequenceOf Connection = empty list, errorReason: String = empty string): void

This operation adds one or more Connections to an existing Call. If necessary, it is also possible to set any TP transmission parameters.

The operation will either set up the entire Connection, or will fail the entire Connection. There is no partial Connection state.

If routing constraints are specified for the Connections in the request, they should match the constraints based on the degree mandated by the appropriate effort statement. If the managed element cannot satisfy the routing criteria, within the boundaries of the effort level requested, an exception is raised.

Operation exceptions:

- 1) NOT_IMPLEMENTED – Raised when the entire operation is not supported, or the operation with the specified input parameters is not supported.
- 2) INTERNAL_ERROR – Raised when the operation has resulted in an internal error.
- 3) INVALID_INPUT – Raised when the operation contains an input parameter that is syntactically incorrect or identifies an object of the wrong type or is out of range.
- 4) ENTITY_NOT_FOUND – Raised when any specified object instance does not exist.
- 5) USERLABEL_IN_USE – Raised when the user label uniqueness constraint is not met; the specified user label is currently being used.
- 6) UNABLE_TO_COMPLY – Raised when the operation cannot respond to the request.
- 7) UNSUPPORTED_ROUTING_CONSTRAINTS – Raised when the operation is unable to satisfy the requested routing constraints.
- 8) OBJECT_IN_USE – Raised when the object identified in the request is currently in use.
- 9) CAPACITY_EXCEEDED: Raised when the request will result in resources being created or activated beyond the current capacity.

Parameter description:

connectionCreateDataList

/ in */* A list of parameters defining the Connection(s) to be created in support of the Call.

connectionRouteRearrangementAllowed

/ in */* Defines whether any existing Connections can be rearranged in order to meet the request.

False = rearrangement not allowed; True = rearrangement allowed.

createdConnections

/ out */* A list of Connections.

errorReason

/ out */* Specifies the creation error(s) if any.

Pre-conditions:

None

Post-conditions:

The operation has created the connections specified, or no connections have been created. Connections are not partially created.

9.3.3.3 removeConnections (connectionNameList: ObjectIdListType = empty list, modifiedCall: Call = NA): void

This operation removes one or more supporting Connections from an existing Call. If necessary, it is also possible to set any TP transmission parameters.

Operation exceptions:

Standard

Parameter description:

connectionNameList

/ in */* The list of Connections to be removed.

modifiedCall

/ out */* The modified Call object.

Pre-conditions:

None

Post-conditions:

The connections specified are removed or an exception has been raised.

9.3.3.4 setAlarmReporting (activated: Boolean = NA): void

This operation activates or de-activates alarm reporting on the Call.

Operation exceptions:

Standard

Parameter description:

activated

/ in */* Defines whether alarm reporting shall be activated = "true" or de-activated = "false".

9.4 CallCreateData

This object class represents the read-create attributes that are required to create a Connection. This object will be passed as one of the parameters in the establishCall operation or in the addConnections operation.

Object Lifecycle: This is part of a Call creation operation. Lifecycle is limited to the duration of the operation.

Object Creation Notification Issued? No

Object Deletion Notification Issued? No

9.4.1 Attributes

9.4.1.1 name: ObjectIdType = NA

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.2 userLabel: String = empty string

This attribute represents a provisionable, user-friendly name for the Call object to be created. It may be initialized to the value in the name attribute by the NE; however, the userLabel attribute is owned and may be set by the EMS.

This attribute may be unique amongst all instances of this object in the control plane.

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.3 aEnd: CallEndType = NA

This parameter identifies the source end point of the Call. The value of this parameter shall be in one of the following name types:

- SNPP;
- SNP;
- TRI Name, which identifies either a TRI or a Group-TRI Name.

If the new Call shall have multiple Connections, then only SNPP or Group-TRI is allowed.

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.4 **zEnd: CallEndType = NA**

This parameter identifies the sink end point of the Call. The value of this parameter shall be in one of the following name types:

- SNPP;
- SNP;
- TRI Name, which identifies either a TRI or a Group-TRI Name.

If the new Call shall have multiple Connections, then only SNPP or Group-TRI is allowed.

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.5 **callParameter: CallParameterType = NA**

This parameter shall identify the requested thresholds (wrt. the number of Connections) of the new Call:

- Degraded Threshold;
- Severely Degraded Threshold.

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.6 **diversityData: DiversityDataType = NA**

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.7 **revertive: Boolean = TRUE**

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.8 **recovery: RecoveryType = NONE**

Allowed values are none (0), restorable (1), protected (2).

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.1.9 asapPointer: ObjectIdType = Null

This attribute indicates the assignment of an ASAP to the Call.

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.4.2 Associations

9.4.2.1 connectionRelatedData [0..*]: Unnamed Association (Unnamed Supplier Role) to ConnectionCreateData

Readable by EMS? No

Writeable by EMS? Yes

Invariant? Yes

Notification Type? None

9.5 CallDetailRecord

The Call Detail Record provides billing information for a specific Call.

Object Lifecycle: Created when the Call is established and explicitly deleted by the EMS after the Call is ended.

Object Creation Notification Issued? No

Object Deletion Notification Issued? No

9.5.1 Attributes

9.5.1.1 callId: OctetString = NA

This attribute represents the unique identifier of the Call.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.5.1.2 startTime: TimeOfDayType = NA

Provides tracking of the duration of a Call sourced by the Call Controller.

Readable by EMS? Yes

Writeable by EMS? No

Invariant? Yes

Notification Type? None

9.5.1.3 stopTime: TimeOfDayType = NA

Provides tracking of the duration of a Call sourced by the Call Controller.

Readable by EMS? Yes

Writeable by EMS? No