INTERNATIONAL STANDARD

ISO/IEC 16500-8

First edition
1999-12-15

Information technology — Generic digital audio-visual systems —

Part 8: Management architecture and protocols

Technologies de l'information — Systèmes audiovisuels numériques génériques —

Partie 8: Architecture et protocoles de gestion

Reference number
ISO/IEC 16500-8:1999(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 734 10 79
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC 16500 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 16500-8 was prepared by DAVIC (Digital Audio-Visual Council) and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

ISO/IEC 16500 consists of the following parts, under the general title *Information technology — Generic digital audio-visual systems*:

— *Part 1: System reference models and scenarios*

— *Part 2: System dynamics, scenarios and protocol requirements*

— *Part 3: Contours: Technology domain*

— *Part 4: Lower-layer protocols and physical interfaces*

— *Part 5: High and mid-layer protocols*

— *Part 6: Information representation*

— *Part 7: Basic security tools*

— *Part 8: Management architecture and protocols*

— *Part 9: Usage information protocols*

Annex A of this part of ISO/IEC 16500 is for information only.

Introduction

ISO/IEC 16500 defines the minimum tools and dynamic behavior required by digital audio-visual systems for end-to-end interoperability across countries, applications and services. To achieve this interoperability, it defines the technologies and information flows to be used within and between the major components of generic digital audio-visual systems. Interoperability between these components and between individual sub-systems is assured through specification of tools and specification of dynamic systems behavior at defined reference points. A reference point can comprise one or more logical (non-physical) information-transfer interfaces, and one or more physical signal-transfer interfaces. A logical interface is defined by a set of information flows and associated protocol stacks. A physical interface is an external interface and is fully defined by its physical and electrical characteristics. Accessible reference points are used to determine and demonstrate compliance of a digital audio-visual subsystem with this international standard.

A summary of each part follows.

ISO/IEC 16500-1 (DAVIC 1.3.1a Part 2) defines the normative digital audio-visual systems technical framework. It provides a vocabulary and a Systems Reference Model, which identifies specific functional blocks and information flows, interfaces and reference points.

ISO/IEC 16500-2 (DAVIC 1.3.1a Part 12) defines system dynamic behavior and physical scenarios. It details the locations of the control functional entities along with the normative protocols needed to support the systems behavior. It is structured as a set of protocol walk-throughs, or "*Application Notes*", that rehearse both the steady state and dynamic operation of the system at relevant reference points using specified protocols. Detailed dynamics are given for the following scenarios: video on demand, switched video broadcast, interactive broadcast, and internet access.

ISO/IEC 16500-3 (DAVIC 1.3.1a Part 14) provides the normative definition of DAVIC Technology Contours. These are strict sets of Applications, Functionalities and Technologies which allow compliance and conformance criteria to be easily specified and assessed. This part of ISO/IEC 16500 contains the full details of two contours. These are the Enhanced Digital Broadcast (EDB) and Interactive Digital Broadcast (IDB). ISO/IEC 16500-3 specifies required technologies and is a mandatory compliance document for contour implementations.

ISO/IEC 16500-4 (DAVIC 1.3.1a Part 8) defines the toolbox of technologies used for lower layer protocols and physical interfaces. The tools specified are those required to digitize signals and information in the Core Network and in the Access Network. Each tool is applicable at one or more of the reference points specified within the Delivery System. In addition a detailed specification is provided of the physical interfaces between the Network Interface Unit and the Set Top Unit and of the physical interfaces used to connect Set Top Boxes to various peripheral devices (digital video recorder, PC, printer). The physical Delivery System mechanisms included are copper pairs, coaxial cable, fiber, HFC, MMDS, LMDS, satellite and terrestrial broadcasting.

ISO/IEC 16500-5 (DAVIC 1.3.1a Part 7) defines the technologies used for high and mid-layer protocols for ISO/IEC 16500 digital audio-visual systems. In particular, this part defines the specific protocol stacks and requirements on protocols at specific interfaces for the content, control and management information flows.

ISO/IEC 16500-6 (DAVIC 1.3.1a Part 9) defines what the user will eventually see and hear and with what quality. It specifies the way in which monomedia and multimedia information types are coded and exchanged. This includes the definition of a virtual machine and a set of APIs to support interoperable exchange of program code. Interoperability of applications is achieved, without specifying the internal design of a set top unit, by a normative Reference Decoder Model which defines specific memory and behavior constraints for content decoding. Separate profiles are defined for different sets of multimedia components.

ISO/IEC 16500-7 (DAVIC 1.3.1a Part 10) defines the interfaces and the security tools required for an ISO/IEC 16500 system implementing security profiles. These tools include security protocols which operate across one or both of the defined conditional access interfaces CA0 and CA1. The interface CA0 is to all security and conditional access functions, including the high speed descrambling functions. The interface CA1 is to a tamper resistant device used for low speed cryptographic processing. This cryptographic processing function is implemented in a smart card.

ISO/IEC 16500-8 (DAVIC 1.3.1a Part 6) specifies the information model used for managing ISO/IEC 16500 systems. In particular, this part defines the managed object classes and their associated characteristics for managing the access network and service-related data in the Delivery System. Where these definitions are taken from existing standards, full reference to the required standards is provided. Otherwise a full description is integrated in the text of this part. Usage-related information model is defined in ISO/IEC 16500-9.

ISO/IEC 16500-9 (DAVIC 1.3.1a Part 11) specifies the interface requirements and defines the formats for the collection of usage data used for billing, and other business-related operations such as customer profile maintenance. It also specifies the protocols for the transfer of Usage Information into and out of the ISO/IEC 16500 digital audio-visual system. In summary, flows of audio, video and audio-visual works are monitored at defined usage data collection elements (e.g. servers, elements of the Delivery System, set-top boxes). Information concerning these flows is then collected, processed and passed to external systems such as billing or a rights administration society via a standardised usage data transfer interface.

Additional Information

ISO/IEC TR 16501 is an accompanying Technical Report. Further architectural and conformance information is provided in other non-normative parts of DAVIC 1.3.1a (1999). A summary of these documents is included here for information.

ISO/IEC TR 16501 (DAVIC 1.3.1a Part 1) provides a detailed listing of the functionalities required by users and providers of digital audio-visual applications and systems. It introduces the concept of a contour and defines the IDB (Interactive Digital Broadcast) and EDB (Enhanced Digital Broadcast) functionality requirements which are used to define the normative contour technology toolsets provided in ISO/IEC 16500-3.

DAVIC 1.3.1a Parts 3, 4 and 5 are DAVIC technical reports. They provide additional architectural and other information for the server, the delivery-system, and the Service Consumer systems respectively. Part 3 defines how to load an application, once created, onto a server and gives information and guidance on the protocols transmitted from the set-top user to the server, and those used to control the set-up and execution of a selected application. Part 4 provides an overview of Delivery Systems and describes instances of specific DAVIC networked service architectures. These include physical and wireless networks. Non-networked delivery (e.g. local storage physical media like discs, tapes and CD-ROMs) are not specified. Part 5 provides a Service Consumer systems architecture and a description of the DAVIC Set Top reference points defined elsewhere in the normative parts of the specification.

DAVIC 1.3.1a Part 13 is a DAVIC technical report, which provides guidelines on how to validate the systems, technology tools and protocols through conformance and / or interoperability testing.

Information technology — Generic digital audio-visual systems — Part 8: Management architecture and protocols

1 Scope

This part of ISO/IEC 16500 describes the management architecture and protocol for managing the DAVIC System. The DAVIC system management architecture is the TMN-based (Telecommunication Management Network) architecture defined in ITU-T Recommendation M.3010. This management architecture allows for automatic administration, configuration, monitoring, billing, and maintenance of the Service Provider System, the Delivery System, and the Consumer System over the DAVIC S5 flows. In accordance with DAVIC 1.3.1a Part 4 and ISO/IEC 16500-5, the content of these flows is based on either the CMIP protocol or the SNMP protocol, depending on the complexity of the underlying subsystem or the preference of the network provider. SNMP MIBs (Management Information Base) for managing the STU and the Server have been defined in ISO/IEC 16500-5. Usage related information models have also been defined in ISO/IEC 16500-9 using both CMISE and SNMP. For managing the ATM, SDH/SONET components of the Core Network of the DAVIC Delivery System, CMISE models defined in ITU-T and ATM Forum are recommended in ISO/IEC 16500-5 subclause 10.5.3. Taking into account that the current DAVIC specification for the Access Network is ATM-based [1], this Part of ISO/IEC 16500 defines CMISE information models for managing the DAVIC Access Network and service-related data in the Delivery System. In the future, if DAVIC specifications allow alternative access technologies (such as IP-based), additional system management information models (such as SNMP MIB) may be needed. The management of the coexistence of different access network technologies, using different protocols, could be achieved through interworking tools (i.e., interworking between CMIP and SNMP) as described in subclause 6.4.

Tutorial information on a protocol-independent modeling technique is also provided in an informative annex.

This part of ISO/IEC 16500 is organized as follows:

Management Architecture	Clause 6 describes the TMN-based management architecture for DAVIC system management.
Access Network Information Model	Clause 7 defines a CMISE information model for the management of the Access Network.
Service Related Control Information Model	Clause 8 defines a CMISE information model for the management of service-related data in the Delivery System.
Informative Annex A: Protocol Independent Modeling Techniques	This informative annex provides tutorial information of management protocol independent modeling technique.

This part of ISO/IEC 16500 refers to system architecture and interface information that are described in ISO/IEC 16500 parts –1, -5 and –9 and in DAVIC 1.3.1a Part 4: *Delivery system architecture and interface*.

2 Normative References

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 16500. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/IEC 16500 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau (TSB) maintains a list of currently valid ITU-T Recommendations.

[1] ATM-based Access Network includes Non-ATM HFC (i.e., Non-ATM at A1 and ATM at A4).

2.1 ISO/IEC and ITU-T normative references

- ITU-T Recommendation G.774, *SDH Management Model for the Network Element View*, 1992 and Corrigendum, 1996.
- ITU-T Recommendation G.774, Corrigendum, 1996.
- ITU-T Recommendation G.774-01, *SDH Performance Monitoring for the Network Element View*, 1994 and Corrigendum, 1996.
- ITU-T Recommendation G.774-06, *SDH Unidirectional Performance Monitoring for the Network Element View*, 1996.
- ITU-T Recommendation I.751, *Asynchronous Transfer Mode (ATM); Management of the network element view.*
- ITU-T Recommendation M.3100, *Generic Network Information Model*, 1995.
- ITU-T Recommendation Q.811 (1993), *Lower layer protocol profiles for the Q3 interface.*
- ITU-T Recommendation Q.812 (1993), *Upper layer protocol profiles for the Q3 interface.*
- ITU-T Recommendation Q.821, *Stage 2 and 3 Functional Descriptions for the Q3 Interface – Alarm Surveillance*, 1992.
- ITU-T Recommendation Q.822, *Stage 2 and 3 Functional Descriptions for the Q3 Interface - Performance Monitoring*, 1994.
- ITU-T Recommendation X.680 (1994) | ISO/IEC 8824-1: 1995, *Information technology - Abstract Syntax Notation 1 (ASN.1): Specification of basic notation.*
- ITU-T Recommendation X.690 (1994), *Information technology - ASN.1Encoding Rules: Specification of BER, CER, DER.*.
- CCITT Recommendation X.700 (1992), *Management framework for Open Systems Interconnection (OSI) for CCITT applications.*
- CCITT Recommendation X.701 (1992), *Information technology – Open Systems Interconnection – Systems management overview.*
- CCITT Recommendation X.711 (1991), *Common management information protocol specification for CCITT applications.*
- ITU-T Recommendation X.721 | ISO/IEC 10165-2:1992, *Information technology - Open Systems Interconnection - Structure of management information: Definition of management information.*
- ITU-T Recommendation X.722 | ISO/IEC 10165-4:1992, *Information technology - Open Systems Interconnection - Structure of management information: Guidelines for the definition of managed objects.*
- ITU-T Recommendation X.745, *Test management function* (1993).

2.2 Additional normative references

The following are additional normative references to the extent specified in Table 7-1.

- ANSI T1.240, *Telecommunications - Operations, Administration, Maintenance and Provisioning (OAM&P) - Generic Network Information Model for Interfaces between Operations Systems and Network Elements*, 1997
- ANSI T1.247, *Telecommunications - Operations, Administration, Maintenance, and Provisioning (OAM&P) Communications - Performance Management Functional Area Services and Information Model for Interfaces Between Operations Systems and Network Elements*, 1997
- ATMF af-nm-0027.000, *CMIP Specification for the M4 Interface*, September, 1995 {#6} available at ftp://ftp.atmforum.com/pub/approved-specs/ af-nm-0027.000.pdf

- ATMF af-nm-0071.000, *Network Management: AAL Management for the M4 "NE View" Interface*, January, 1997 {#6} available at ftp://ftp.atmforum.com/pub/approved-specs/ af-nm-0071.000.pdf
- ATMF af-nm-0072.000, *Circuit Emulation Service Interworking Requirements, Logical and CMIP MIB*, January, 1997 {#6} available at ftp://ftp.atmforum.com/pub/approved-specs/ af-nm-0072.000.pdf
- Bellcore GR-1114-CORE, *Generic Operations Interface Requirements : ATM Information Model*, Issue 3, 1996

3 Definitions

This clause defines new terms, and the intended meaning of certain common terms, used in this part of ISO/IEC 16500. Annex A of ISO/IEC 16500-1 defines additional terms and, in some cases, alternative interpretations that are appropriate in other contexts. For convenience, the normative definitions below are included in the annex.

3.1. Actor: A person or system component who interacts with the system as a whole and who provides stimulus which invoke actions.

3.2. Class: Describes a group of Objects with similar properties and behavior.

3.3. Class Diagram: Shows the existence of classes and their relationships in the logical view of the system.

3.4. Class and Object Modeling: Used to define business objects and application architecture. Shows how classes interoperate dynamically to offer use cases and application behavior.

3.5. Component Diagram: Shows the dependencies between software components.

3.6. Computational Model: Gives a description of a system in a graphical form highlighting Objects and their interfaces, as such it is similar to the OMT and UML notion of a class Diagram. An Object Distributed processing (ODP) concept.

3.7. Deployment Modeling: Allows you to model how your application is mapped to a distributed deployment network.

3.8. Domain Analysis: An iterative process that focuses on what is in the system but not on how it is implemented. Domain Analysis builds a language/system independent description of the domain.

3.9. Event Trace Diagram: Interaction Diagram or Message Trace Diagram.

3.10. Functional Model (OMT): Shows how objects in the system interact.

3.11. Interaction Diagram: Similar in principle to collaboration diagrams. Show the interaction between objects over time.

3.12. Message Trace Diagram: Interaction Diagram or Event Trace Diagram.

3.13. Method: A systematic way to achieve a particular goal.

3.14. Methodology: A coherent, integrated set of methods from which a coherent sub-set can be selected for particular applications. A methodology should contain at least four components:

1. a conceptual model of constructs essential to the problem,
2. a set of procedures suggesting the direction and order to proceed,
3. a series of guidelines identifying things to be avoided, and
4. a collection of evaluative criteria for assessing the quality of the product.

3.15. Methodology framework: a way, or structure, that supports a number of methods and languages to be used together when developing a system.

3.16. Network Element (NE): A component of a telecommunications network, whose function is to provide communications services to users. These functions include information transport, adaptation, switching and control.

3.17. Object: A representation of a system component that has both state and behavior.

3.18. Object Collaboration Diagram: Provides a snapshot of a systems state and is used to emphasize relationships.

3.19. Object Model: Shows the static relationships between classes. An object model is described using a class diagram. The Object Model defines the behavior required by the various classes to ensure that the use cases and business rules are supported correctly.

3.20. Operations Support System (OS): A system whose main function it is to run applications that manage network elements, networks and services.

3.21. Package: A nested name space in a model, which contains classes.

3.22. Scenario Diagram: Shows how the Objects in a system interact over time.

3.23. State Transition Diagram: Shows the state of a given context, the events that cause a transition from one state to another and the actions that result.

3.24. Telecommunications Management Network (TMN): A network consisting of Operations Support Systems (OS) and data networking facilities whose purpose it is to provide management capabilities for telecommunications networks, network elements, resources and services.

3.25. Technique: A way that a method is realized, or implemented.

3.26. Tool: A software item, which automates or supports the use of a particular method or language. In system/software engineering, a step-by-step, formalized, manual or automated, process for solving an engineering problem.

3.27. Use Case Diagram: Shows the system's main Use Cases and the Actors that interact with them.

3.28. Use Case Modeling: A requirement analysis technique. Use Cases capture the functional requirements of a system.

4 Acronyms and abbreviations

This clause defines the acronyms and abbreviations used in this part of ISO/IEC 16500. Annex B of ISO/IEC 16500-1 defines acronyms and abbreviations used within ISO/IEC 16500.

AAL	ATM Adaptation Layer
ACTS	Advanced Communications Telematics Services
ADSL	Asymmetric Digital Subscriber Line
ANSI	American National Standards Institute
ASN.1	Abstract Syntax Notation 1
ATM	Asynchronous Transfer Mode
AU4	Administrative Unit - 4
AU44C	Administrative Unit - 44C
AUG	Administrative Unit Group
BCU	Broadcast Control Unit
CAGR	Customers' Anticipate Growth Rate
CAP	Carrierless Amplitude and Phase Modulation
CASE	Computer Aided Software Engineering
CATV	Cable TV
CCP	Channel Change Protocol
CFP	Call for Proposals
CMIS	Common Management Information Service
CMISE	Common Management Information Service Element
CMIP	Common Management Information Protocol
CORBA	Common Object Request Broker Architecture
CPE	Customer Premises Equipment
CPN	Customer Premises Network
CTP	Connection Termination Point
DAB	Digital Audio Broadcasting
DAVIC	Digital Audio Visual Council
DBS	Digital Broadcast Satellite
DBR	Deterministic Bit Rate
DII	DAVIC Interface Initialization
DIIP	DAVIC Interface Initialization Protocol

DS3	Digital Signal - Level 3
DSM-CC	Digital Storage Media - Command & Control
DTTB	Digital Terrestrial Television Broadcasting
DVB	Digital Video Broadcasting
EDI	Electronic Data Interchange
EMC	Electromagnetic Compatibility
EMMA	European Multimedia experiments in ATM environment
EMS	Element Manager System
FCC	Federal Communications Committee
FITL	Fiber Into The Loop
FR	Frame Relay
FSN	Full Service Network
FTP	File Transfer Protocol
FTTB	Fiber To The Building
FTTC	Fiber To The Curb
FTTH	Fiber To The Home
GDMO	Guidelines for the Definition of Managed Object
HFC	Hybrid Fiber Coax
HMI	Human Machine Interface
HFTTB	Hybrid Fiber To The Building
HTML	Hyper Text Mark up Language
IDL	Interface Definition Language
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internal Engineering Task Force
IN	Intelligent Network
INVOD	Intelligent Near Video On Demand
ISP	Internet Service Provider
ITU	International Telecommunication Union
ITU-T	ITU - Telecommunications Standardization Sector
JPEG	Joint Photographic Experts Group
LAN	Local Area Network
LEWS	Local Exploitation Work station
LMDS	Local Multipoint Distribution Service
MIB	Management Information Base
MF	Management Function
MHEG	Multimedia and Hypermedia Information coding Experts Group
MMC	Multimedia Conferencing
MMS	Multimedia Conference Services
MO	Managed Object
MOC	Managed Object Class
MOD	Movies On Demand
MPEG	Motion Pictures Experts Group
MS	Multiplex Section
MVDS	Multimedia Video on Demand System
NC	Network Computer
NMF	Network Management Forum
NMS	Network Management System
NOD	News On Demand
NT	Network Termination
NVOD	Near Video On Demand
PDH	Plesiochronous Digital Hierarchy
PLCP	Physical Layer Conversion Protocol
RF	Radio Frequency
RS	Regenerator Section
QAM	Quadrature Amplitude Modulation
OAN	Optical Access Network
OFDM	Orthogonal Frequency Division Multiplexing
OLT	Optical Line Termination
OMG	Object Modeling Group

OMT	Object Modeling Technique
ONU	Optical Network Unit
ONP	Open Network Provision
OO	Object Oriented
OOAD	Object Oriented Analysis and Design
QPSK	Quaternary Phase Shift Keying
OSI	Open System Interconnection
PC	Personal Computer
PCR	Peak Cell Rate
PIN	Personal Identification Number
PNO	Public Network Operator
PON	Passive Optical Network
POTS	Plain Old Telephony System
PPI	Pay Per Information unit
PPP	Pay Per Page
P-PSN	Public Packet Switched Network
PPT	Pay Per Time
PPV	Pay Per View
PSN	Public Switched Network
PSTN	Public Switched Telephone Network
PVC	Permanent Virtual Connection
QoS	Quality of Service
RDN	Relative Distinguished Name
RSVP	Resource ReServations Protocol
SBR	Statistical Bit Rate
SCR	Sustainable Cell Rate
SDB	Switched Digital Broadcast
SDH	Synchronous Digital Hierarchy
SDV	Switched Digital Video
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical Network
SPI	Synchronous Physical Interface
SPVC	Semi-Permanent Virtual Connection
STB	Set Top Box
STU	Set-Top Unit
SVC	Switching Virtual Connection
TC	Transmission Convergence
TCP/IP	Transmission Control Protocol/Internet Protocol
TMN	Telecommunication Management Network
TTP	Trail Termination Point
VC	Virtual Channel
VC4	Virtual Container 4
VC44C	Virtual Container 44C
vcCTP	Virtual Channel Connection Termination Point
VCI	Virtual Channel Identifier
UDP	Usage Data Protocol
VDSL	Very high bit rate Digital Subscriber Loop
UML	Unified Modeling Language
VP	Virtual Path
VPI	Virtual Path Identifier
URL	Uniform Resource Locator
UNI	User-Network Interface
VDT	Video Dial Tone
VOD	Video On Demand
VPN	Virtual Private Network
WWW	World Wide Web

5 Conventions

The style of this part of ISO/IEC 16500 follows the general guidelines of *Guide for ITU-T and ISO/IEC JTC 1 cooperation. Appendix II: Rules for presentation of ITU-T | ISO/IEC common text, (March 1993).*

6 DAVIC System Management Architecture

6.1 Scope of DAVIC System Management Architecture

This document describes the management architecture of a DAVIC system. The DAVIC system management architecture allows for automated administration, monitoring, billing, and maintenance of the Service Provider System, the Delivery System, and the Consumer System over DAVIC S5 flows. These flows may occur within a single DAVIC system or between DAVIC systems. They may also be carried transparently through a DAVIC system in order to reach another DAVIC system. In accordance with DAVIC 1.3.1a Part 4, the content of these flows is based on either the CMIP protocol or the SNMP protocol, depending on the subsystem. Industry trends may encourage the addition of Web-based management or object broker-based management (e.g., CORBA) in future DAVIC specifications.

To deal with the complexity of managing a DAVIC network and its services, the management functionality is considered to be partitioned into logical layers, as defined in the TMN Logical Layered Architecture (LLA) of ITU-T M.3010. A logical layer reflects a particular level of abstraction of the network or service and addresses aspects of management that are associated with that level of abstraction. The following five logical layers are defined:

- Business Management Layer;
- Service Management Layer;
- Network Management Layer;
- Element Management Layer; and
- Network Element Layer.

Management aspects of network elements, networks, and services are supported by applications called operations support functions (OSF). An OSF is a unit of management functionality, which meets a focused set of management goals. An OSF client is typically implemented within one of the four management layers, while its server is implemented in one or more layers below it. To satisfy the TMN architecture, each layer of an OSF must provide sufficient functionality to serve its own purpose and to meet the needs of the layers above it which address the same overall management goals. It is therefore apparent that the Network Element Layer, which is typically implemented in an embedded system, has the broadest scope since it has the potential to serve all four management layers.

It should be noted that logical management layers and OSFs are conceptual in nature, and that the implementation of a management system may combine layers or functions.

6.2 Partitioning

6.2.1 Functional OSF Partitioning

An example of OSF partitioning based upon business, service, network, and element management applications is depicted in Figure 6-1. As shown, some implementations may include business OSFs that are concerned with a total enterprise (i.e., all services and networks) and carry out an overall business goal. Service OSFs are concerned with services offered by one or more networks and will normally perform a customer-interfacing role. Network OSFs are concerned with the management of networks, and Element OSFs with the management of individual elements.

Network OSFs cover the realization of network-based TMN application functions by interacting with Element OSFs. Element and Network OSFs collaborate to manage activities across the network and support “Network” demands of service OSFs. Thus, Element OSFs and Network OSFs share the infrastructure aspects of a telecommunications network, whereas Service and Business OSFs are founded on the financial and service-related practices of an enterprise, which pertain to or depend on the network.

The layering of OSFs based in the reference model shown in Figure 6-1, although widely accepted, should not be regarded as the only possible implementation of DAVIC management systems. Additional or alternative layers may be used to specialise the functionality.

6.2.2 TMN Logical Layer Partitioning

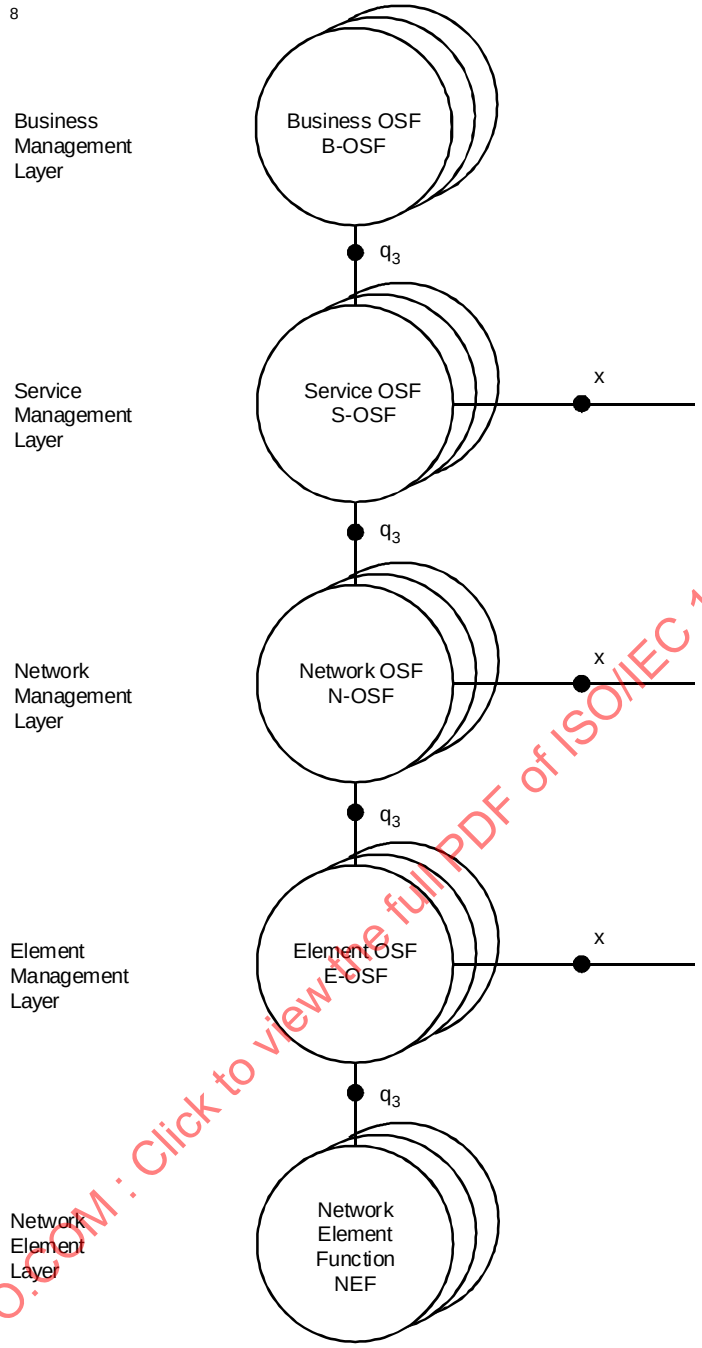
The reference model of an operations system (OS) functional architecture with four management layers is illustrated in Figure 6-2 and described in the following subclauses.

6.2.2.1 Element Management Layer

The element management layer manages each network element on an individual or group basis and supports an abstraction of the functions provided by the network element layer.

The element management layer has the following three principal roles:

- The control and coordination of a subset of network elements on an individual NEF basis. In this role, the element OSFs support interaction between the network management layer and the network element layer by processing the management information being exchanged between network OSFs and individual NEFs. Element OSFs should provide full access to NE functionality.
- The element management layer may also control and coordinate a subset of network elements on collective basis. In this role, element OSFs may also provide a single entity view of a group of NEFs. In addition, the element OSFs may manage the relationships (e.g. connectivity) between NEFs.
- Maintaining statistical, log and other data about elements within its scope of control.



NOTES

1 Additional or alternative layers are permitted.

2 Other interactions may also occur between non-adjacent layers.

Figure 6-1 — Reference Model of OSF Architecture

OSFs in the element management layer interact with OSFs in the same or other layers within the same DAVIC management network through a q3 reference point and in other DAVIC management networks through an x reference point.

The Element Management Layer information model for a DAVIC Access Network is defined in Clause 7. Future DAVIC specifications may define additional Element Management models for the Core Network, the Service Provider System, and the Consumer System.

6.2.2.2 Network management layer

The network management layer has the responsibility for the management of a network as supported by the Element Management layer.

At this layer, functions addressing the management of a wide geographical area are located. Complete visibility of the whole network is typical and, as an objective, a technology independent view will be provided to the service management layer.

The network management layer has the following four principal roles:

1) The control and coordination of the network view of all network elements within its scope or domain, argmenting the Element Management Layer by its knowledge of physical and logical connectivities.
2) The provision, cessation or modification of network capabilities for the support of service to customers.
3) The maintenance of network capabilities.
4) Maintaining statistical, log and other data about the network and interact with the service manager layer on performance, usage, availability, etc.

Thus, the network management layer provides the functionality to manage a network by coordinating activity across the network and supports the “network” demands made by the Service Management Layer. It knows what resources are available in the network, how these are interrelated and geographically allocated and how the resources can be controlled. It has an overview of the network. Furthermore, this layer is responsible for the technical performance of the actual network and will control the available network capabilities and capacity to give the appropriate accessibility and quality of service.

The Element Management and Network Management Layers are traditionally subdivided into the following functionalities, as described in DAVIC 1.3.1a Part 4:

- Network Client Directory Management (not currently addressed).
- Configuration Management (addressed in Clause 7 for the Access Network).
- Fault Management (addressed in Clause 7 for the Access Network).
- Performance Management (not currently addressed).
- Security Management (not currently addressed).
- Usage Data Collection (addressed in ISO/IEC 16500-9).

OSFs in the network management layer interact with OSFs in the same or other layers within the same TMN through a q3 reference point and in other TMNs through an x reference point.

6.2.2.3 Service management layer

Service management is concerned with, and responsible for, the contractual aspects of services that are being provided to customers or available to potential new customers. Some of the main functions of this layer are service order handling, complaint handling and invoicing.

The service management layer has the following four principal roles:

1) customer facing and interfacing with other administrations/ROAs;
2) interacting with service providers;
3) maintaining the statistical data (e.g. QOS);
4) interacting between services.

Customer facing provides the basic point of contact with customers for all service transactions including provision/cessation of service, accounts, QOS, fault reporting, etc.

OSFs in the service management layer interact with OSFs in the same or other layers within the same TMN through a q3 reference point and in other TMNs through an x reference point.

The Service Management layer is responsible for all negotiations and resulting in contractual agreements between a (potential) customer and the service(s) offered to this customer. It is traditionally subdivided into the following functionalities, as described in DAVIC 1.3.1a Part 4:

- ESP Directory Management (not addressed).
- User Directory Management (addressed by Clause 8).
- STU Directory Management (addressed by Clause 8).
- Usage Parameter Collection (addressed by ISO/IEC 16500-9).

The Service Management Layer management information model for DAVIC systems is defined by Clause 8. It addresses a service provider's management needs of SDV and VOD services. It is supplemented by the STU MIB in ISO/IEC 16500-5, Annex A, and the Server MIB in ISO/IEC 16500-5, Annex B. Future DAVIC specifications may define the management of additional service applications.

6.2.2.4 Business management layer

The business management layer has responsibility for the total enterprise. The business management layer comprises proprietary functionality. To prevent access to its functionality, business OSFs do not normally support x reference points. Business OSFs access the information and functionality in the other management layers. The business management layer is included in the TMN architecture to facilitate the specification of capability that it requires of the other management layers.

This layer normally carries out goal setting tasks rather than goal achievement but can become the focal point for action in cases where executive action is called for. This layer is part of the overall management of the enterprise and many interactions are necessary with other management systems.

While the main functions of service and network management layers are the optimal utilization of existing telecommunications resources, those of the business management layer are for the optimal investment and use of new resources.

OSFs in the business management layer interact with OSFs in the same or other layers within the same TMN through a q3 reference point.

The business management layer has the following four principal roles:

1) Supporting the decision-making process for the optimal investment and use of new telecommunications resources;
2) Supporting the management of AO&M related budget;
3) Supporting the supply and demand of AO&M related manpower;
4) Maintaining aggregate data about the total enterprise.

The existing DAVIC management information models do not address the Business Management Layer.

Figure 6.2 below shows a possible assignment of management functionality consistent with the above principles.

6.2.3 DAVIC Management Scenarios

This sub-section provides example management scenarios. These scenarios are aligned with the scenarios in DAVIC 1.3.1a Part 4, Clause 11 but provide more detail concerning the management network architecture and interfaces. It should be noted that these are only example scenarios and the division of functionality and layering of EMS (Element Management System), NMS (Network Management System) and SMS (Service Management System) may vary.

6.2.3.1 Scenario 1

This is a scenario in which the end-to-end multimedia system is partitioned into two management domains based on provider boundaries. One management domain contains the Service Provider who owns the server. The second management domain contains the Delivery System provider who owns the Core Network, Access Network and the STB. These two management domains are managed by separate Management Systems to provide reliability in their portion of the end-to-end system. There is peer-to-peer communication between the two SMSs.

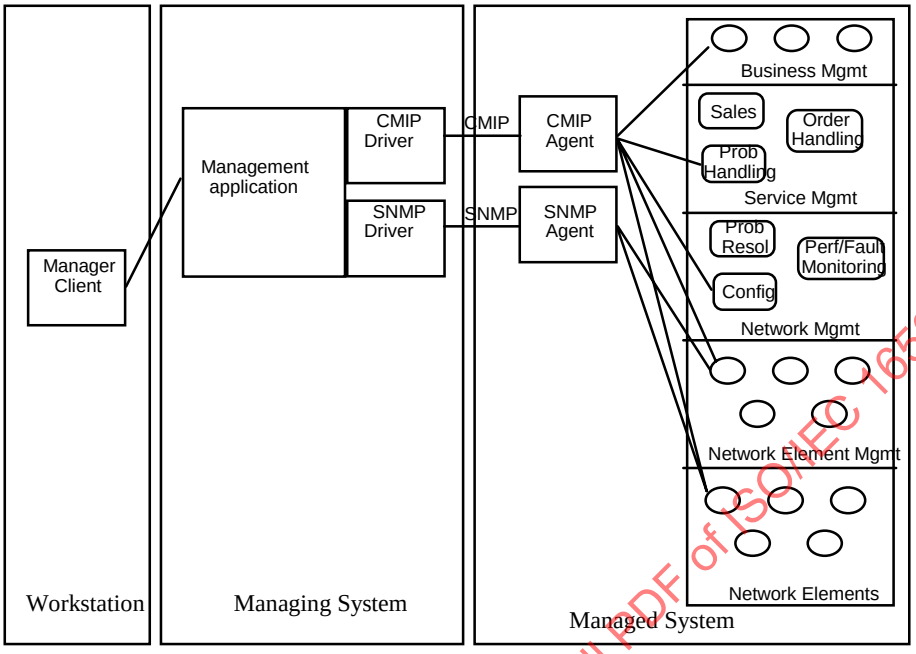


Figure 6-2 — OSF Partitioning Example

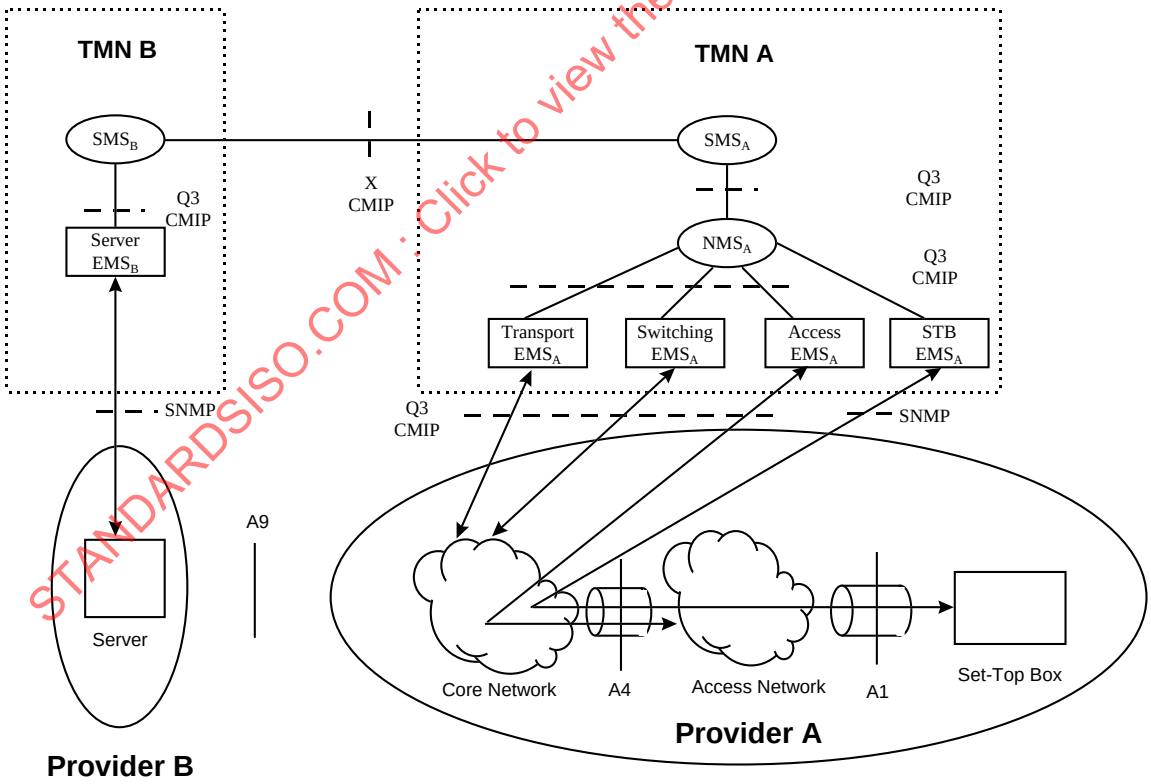


Figure 6-3 — Management Scenario 1

6.2.3.2 Scenario 2

This is a scenario in which the end-to-end multimedia system is partitioned into two management domains based on provider boundaries. One management domain contains the Service Provider who owns the server and the STB. The second management domain contains the network provider who owns the Core Network, and the Access Network. The two domains are managed by separate Network Management Systems. There is peer to peer communication between the two SMSs.

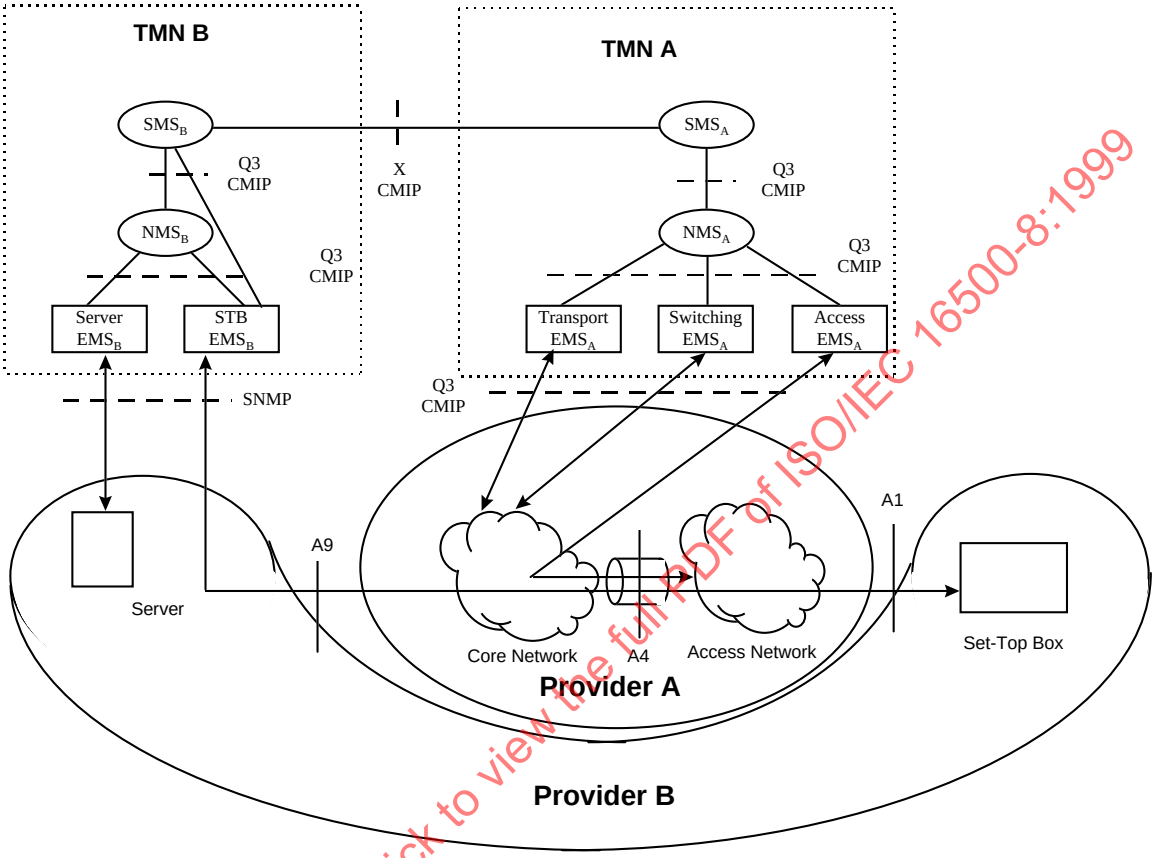


Figure 6-4 — Management Scenario 2

6.2.3.3 Scenario 3

This scenario is one in which the STB is owned by an End Consumer. The end-to-end multimedia system without the STB can now be partitioned into management domains based on the ownership of the several components of the system. The figure identifies one of the possible scenarios.

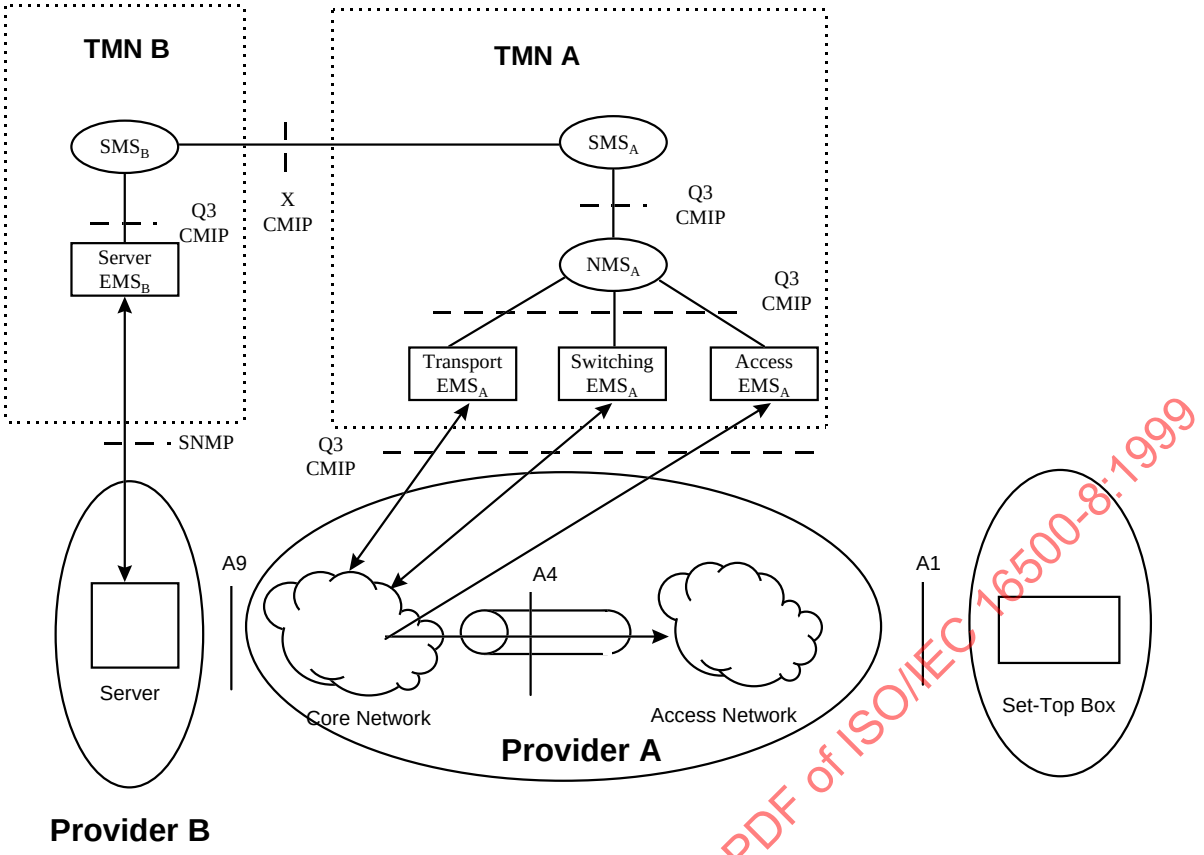


Figure 6-5 — Management Scenario 3

6.2.3.4 Scenario 4

This scenario shows a single enterprise scenario in which one provider owns the entire network. In this case, conceptually only one SMS is required to manage the network to provide reliable service. However to decrease complexity, the network can be partitioned into subnetworks to provide distributed management.

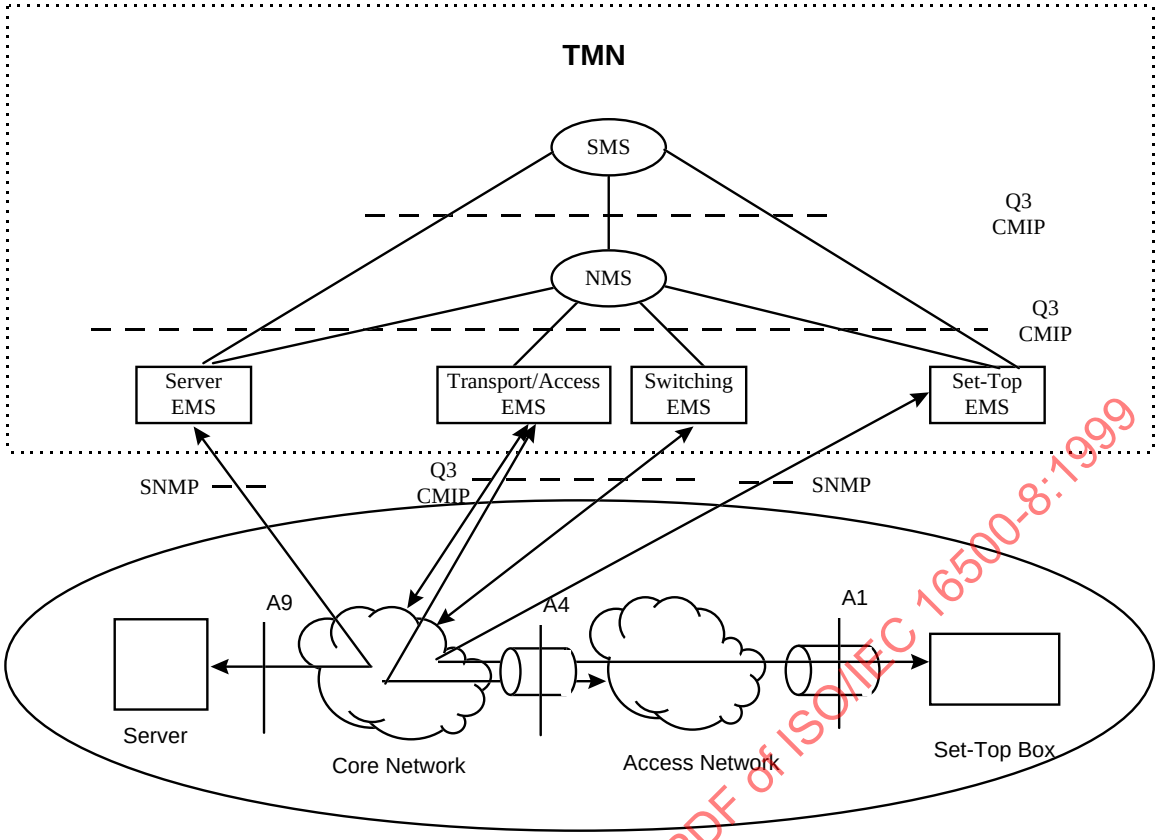


Figure 6-6 — Management Scenario 4

6.2.3.5 Scenario 5

In this scenario the Access Network provider also owns the STB. The end-to-end multimedia system can now be partitioned into management domains based on the ownership of the several components of the system as below.

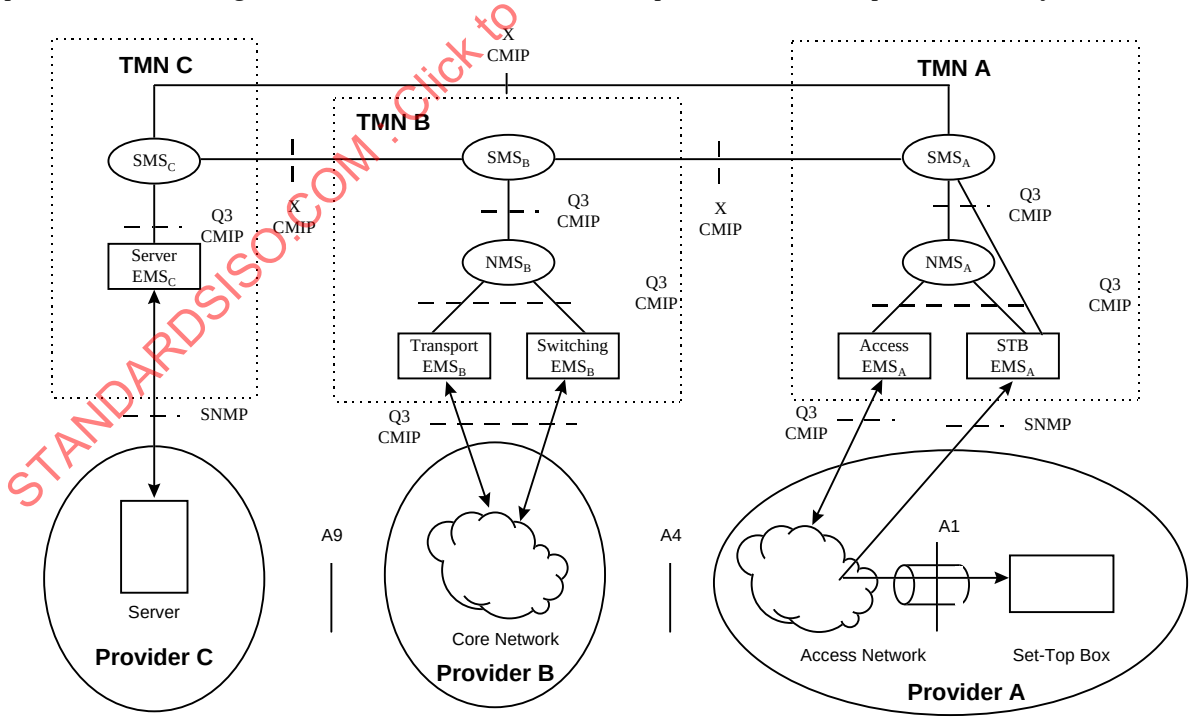


Figure 6-7 — Management Scenario 5

6.2.3.6 Scenario 6

In this scenario all resources belong to the network provider, however the network provider is selling capacity to service providers. Interactions, therefore, occur at the service level but all flowing within the DAVIC network are controlled by the network provider.

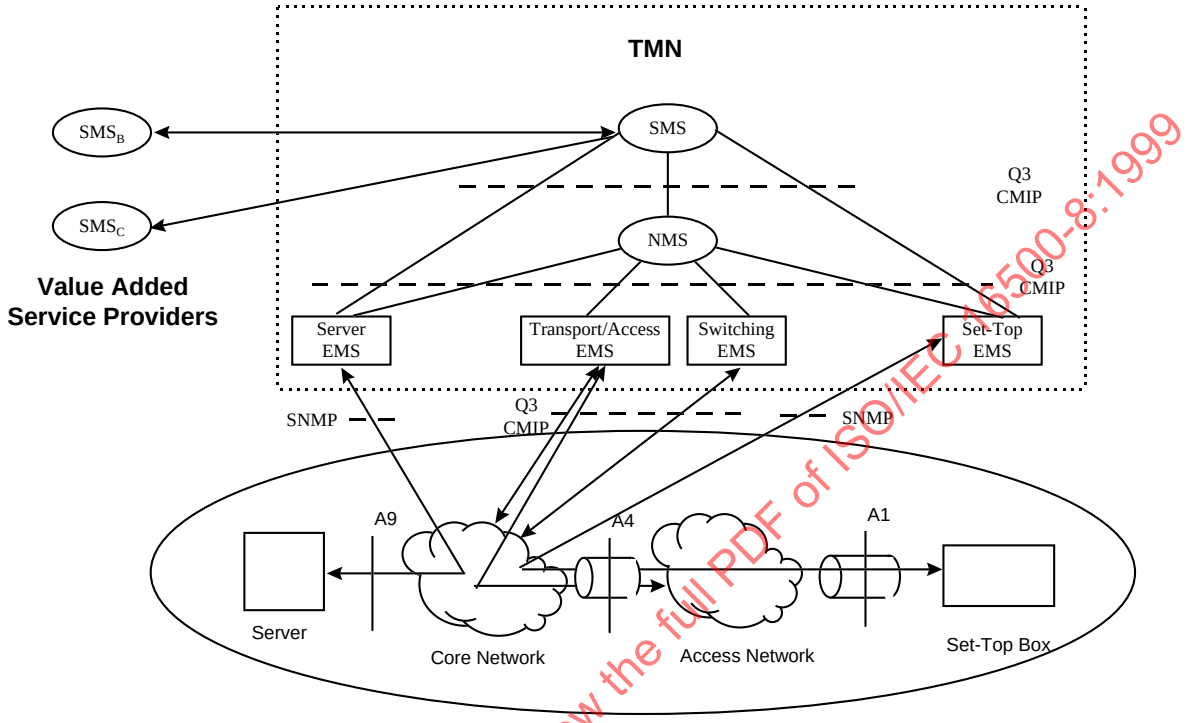


Figure 6-8 — Management Scenario 6

6.3 Interworking Scenarios Between CMIP and SNMP Based Management

This sub-section describes interworking options between CMIP and SNMP based management systems. Three possible scenarios are described. In addition it is noted that semantic integration within the managing system may be possible via an ORB (e.g. CORBA) but this semantic integration is not mandated by DAVIC and the ORB interface is not a DAVIC interface.

The three options are:

1. Integration through the Human-Machine Interface (HMI). The idea is to centralize on a single X-Window terminal the HMIs of several already implemented network managers.
2. Dual protocol stacks. The manager possesses two protocol stacks: a CMIP stack for communicating with CMIP agents, a SNMP stack for communicating with SNMP agents. A common API, such as X/Open XMP can be used over those two stacks.
3. CMIP-to-SNMP proxy. This proxy acts in a manager role regarding the SNMP agents, and in a agent role regarding the manager.

    - The CMIP MIB viewed by the manager is the translation of the SNMP MIB of the SNMP agents.
    - The proxy also does CMIP service emulation over SNMP i.e. it converts traps and/or polling operations into CMIP notifications and also translates scope and filter requests to individual SNMP operations and returns the correct CMIP results to the upstream manager.

As an option to facilitate the programming of management applications in a management system, “semantic” integration may be provided. The goal is to provid the management applications the same procedure calls to perform a given task on a given network resource type regardless of the fact that a particular network resource is accessed via CMIP or SNMP and whether or not it uses a standardized or a proprietary MIB.

This “semantic” integration could be accomplished by internal implementation of an IDL interface with which the internal applications interact. Beneath the IDL interface operations on managed objects are then invoked over dual protocol stack to the appropriate agent.

6.3.1 Integration through the Human-Machine Interface

In this scenario, every agent that is to be integrated is already managed by a management system, specific to a protocol, and offering a complete Human-Machine Interface, based for instance on X-Window. Bringing all the management capability to a central location is then realized allowing access to all these windows through a single X-Terminal. In the simple cut-through scenario this will result in the user having to adapt to a different look and feel for each window. Tighter integration is possible if the individual management systems, independent of protocol, implement the same look and feel and a consistent set of tools.

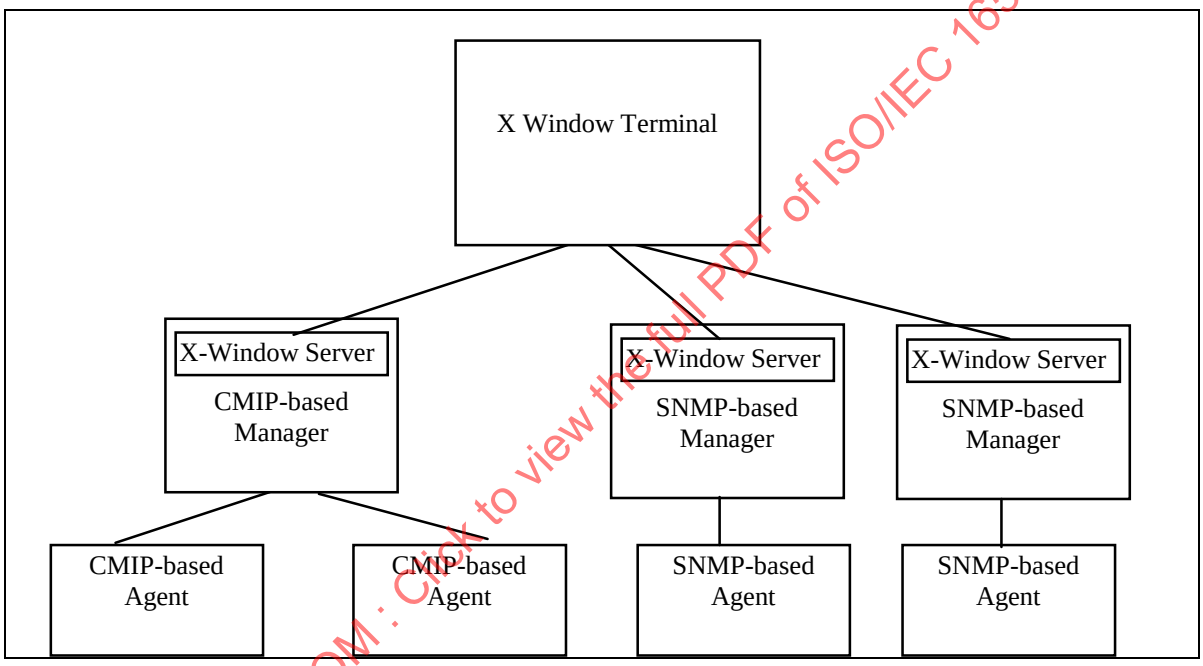


Figure 6-9 — Integration through the Human-Machine Interface

6.3.1.1 Advantages

This architecture offers:

- Ease of implementation
- Ability to integrate both standard and proprietary protocols
- Reuse of existing managers. This is particularly interesting in the case when a managing system, including an HMI and acting as a « black box », is provided by the managed equipment vendor.

6.3.1.2 Disadvantages

The end user remains aware of the multiple management systems. The end user may have to adapt to the « look and feel » of each system. He has to do all the job of integration and coordination between the different manager. For every task he has to do, he has to know and use the tool provided for this task by each manager.

6.3.2 Dual protocol Stacks

In this approach, the manager possesses two protocol stacks : a CMIP stack for communicating with CMIP agents, a SNMP stack for communicating with SNMP agents. Those stacks are accessed and used via different Application Programming Interfaces (APIs). In a refinement of this approach, an API common to both stacks can be defined. An example of such an API is the X/Open Management Protocol API (XMP).

XMP tries, as far as it is possible, to provide the same « C » programming language function calls and data structures for these requests that are common to both protocols (for example, simple Get and Set requests). Of course, those requests that are specific to one protocol correspond to a specific function of the API (for example SNMP GetNext, CMIP filter and scope requests, etc.)

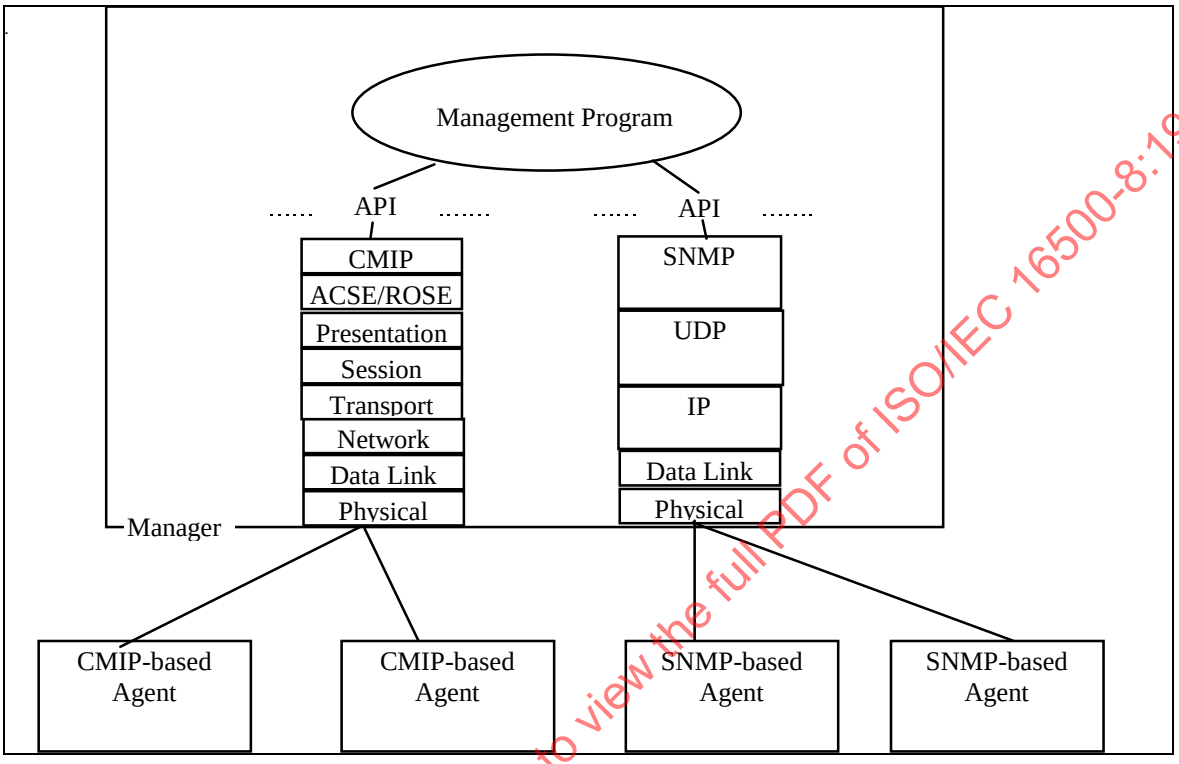


Figure 6-10 — Dual Protocol Stacks Example

6.3.2.1 Advantages

This architecture effectively permits the co-existence of CMIP and SNMP agents.

6.3.2.2 Disadvantages

This scenario doesn't provide real protocol transparency. The management program developer should be aware of which protocol will be used to access the network resources he wants to manipulate. For instance, he will be able to issue GetNext requests only for those resources that will be reached via SNMP.

Should the developer want to emulate a CMIP service like a scope and filter get (i.e. to access a set of resources to verify some properties) over a SNMP table, he would have to download the whole table (using SNMP GetNext). This would cause heavy traffic between the manager and the SNMP agent.

6.3.3 CMIP-to-SNMP Proxy

In this scenario, a CMIP-to-SNMP proxy is used to access a SNMP-based agent. One such proxy is specified by the NMF.

This proxy acts in a manager role with respect to the SNMP agents, and in a CMIP agent role with respect to the manager. The CMIP MIB viewed by the manager is the translation of the SNMP MIB of the SNMP agent. A translation procedure which can be performed almost fully automatic is defined by the NMF. Groups and

RowEntry objects of a SNMP MIB are converted to GDMO objects, SNMP scalar objects are converted to GDMO attributes, SNMP traps are converted to CMIP notifications, etc. All these GDMO objects are registered under a specific registration node, owned by the NMF.

This proxy also does service emulation. For instance:

- If the proxy receives a scope and filter CMIP request from the manager, it will automatically generate the corresponding Get and GetNext requests to the SNMP agent, process the results, filter them and send back to the manager only the required information.
- The proxy will do scheduled polling to emulate CMIP state transition notifications and will convert SNMP poll results into CMIP notifications.
- The proxy will be able to filter the SNMP traps, according to a CMIP Event Forwarding Discriminator set by the manager.

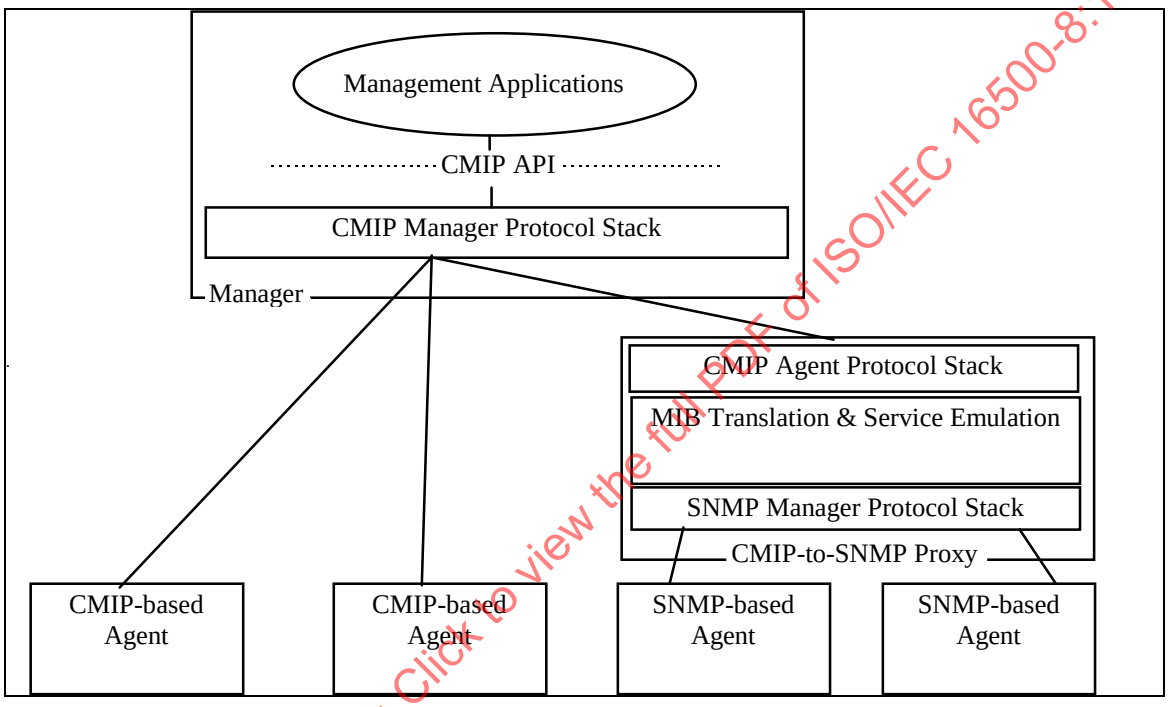


Figure 6-11 — CMIP-to-SNMP Proxy Example

6.3.3.1 Advantages

This architecture provides real protocol transparency. The management application programmer has only to deal with a CMIP API. He can use every service available on a CMIP interface.

The proxy can be located “near” the SNMP agents. The emulation of CMIP services over SNMP will consume high bandwidth only on the short distance between the proxy and the SNMP agents.

6.3.3.2 Disadvantages

This architecture doesn’t provide “semantic” transparency, defined as follows : suppose two different network equipments that have the same network functionality (e.g. both of them are ATM cross-connects), but are provided by different vendors and have different agents. One has a CMIP MIB (say *M1*) and the other a SNMP MIB (say *M2*). With the CMIP-to-SNMP Proxy approach, and in particular with the automated MIB translation approach, SNMP MIB *M2* will be translated to a CMIP MIB (say *M3*). However *M1* and *M3* will be different (different object registrations at least, probably not the same conceptual attributes, etc.). This means that to do the same thing on these two equipments (for instance, to create an ATM cross-connection), the management application developer will have to issue two different CMIP requests.

7 Access Network Management Information Model

7.1 Scope of Access Network Management

This section describes the management of Access Network at the A8 interface in the DAVIC reference model. It provides a high level description of the functions required to manage an access network in the Delivery System of the DAVIC reference model. Aspects such as configuration management, fault management and performance management are addressed. An object model supporting these functions is also presented. Figure 7-1 shows the Delivery System Control Plane Model, taken from ISO/IEC 16500-1.

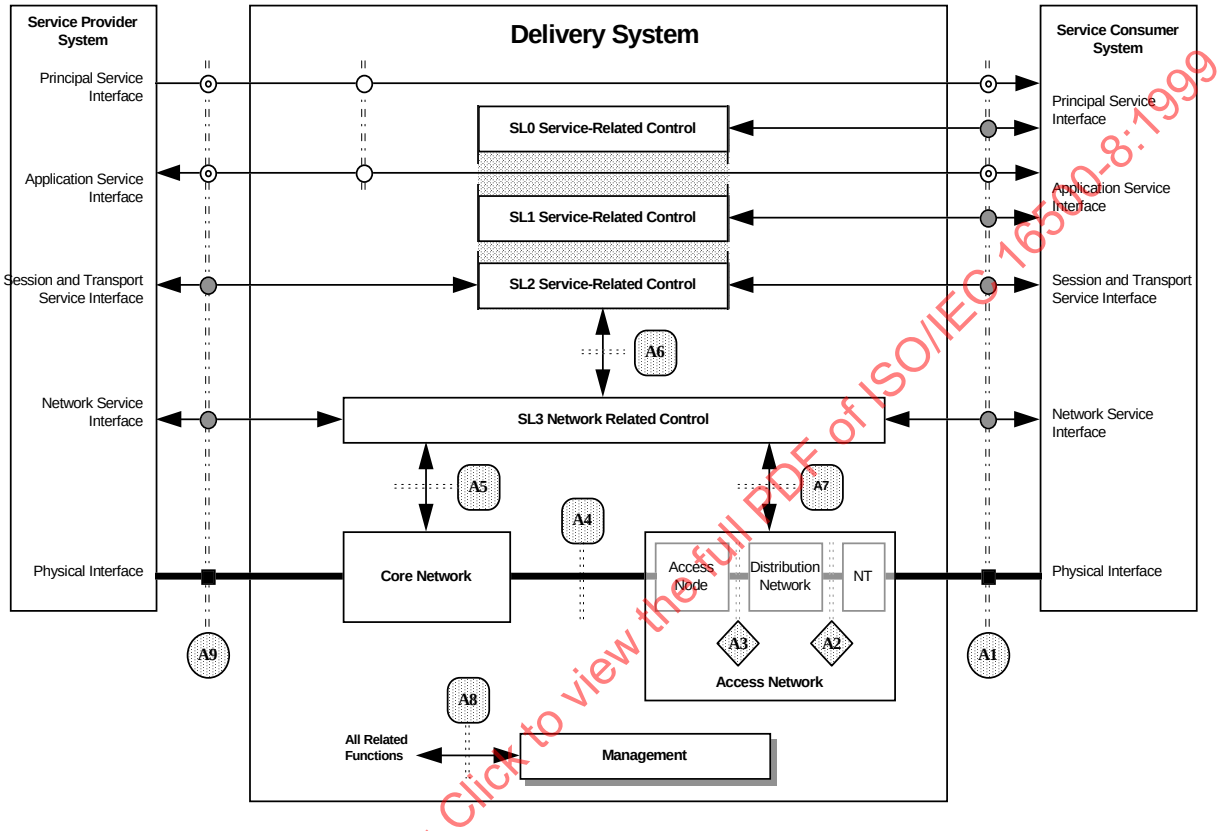


Figure 7-1 — Delivery System Control Plane Model

Services that need to be managed are called broadband services as it is assumed that the core network is a SDH/SONET and ATM based network. Examples of the services considered are Video On Demand (VoD), News On Demand (NoD), Internet Access, etc. The Access Network itself may use various technologies. Currently the technologies that are envisaged for the Access Network are cabled networks such as Hybrid Fiber Coax (HFC), copper networks such as Asymmetric Digital Subscriber Line ADSL and passive optical networks such as FTTC.

In the access network information model, the scope of an Access Network includes an access node (e.g., such as a Host Digital Terminal - HDT) and its subtending distributed components (e.g., such as the Optical Network Units - ONU, Network Terminals - NT, and the fiber, cable or copper pair). It also includes the physical and logical resources within the access node and it subtending components. The resources considered include hardware, software, transport termination points (TPs) of the various transport layers (e.g., SDH or PDH, ATM, DAVIC Information Flow) at the A4 interface (i.e., with the code network) and A1 interfaces (i.e., with the CPEs), intra access network transport TPs, Cross-connections that support nail-up video services. In addition, it also includes logical resources that support the management of the access network.

Clause 7 is organized as follows : first, the access network architecture is considered. This first part sets the framework for the rest of the document. A second part describes the management architecture and functions of the Access Network. These functions are necessary in order to provision and monitor the resource and service

established on the access portion of the network. Lastly, the object model and some simple scenarios are presented in order to illustrate the previous parts.

7.2 Network Delivery Systems Architecture and Interfaces

Figure 7-2 below shows the logical view of the architecture and interface of a Delivery System. The Delivery System being shown is a Cabled Network Delivery System (NDS) defined for DAVIC 1.1. Other NDS and Non-Networked Delivery Systems are for future study. In the figure, no in-house network is considered. In other words the Set Top Box (STB) is directly connected to the Network Termination (NT), i.e. the A1* interface described in DAVIC 1.3.1a Part 4 is nil.

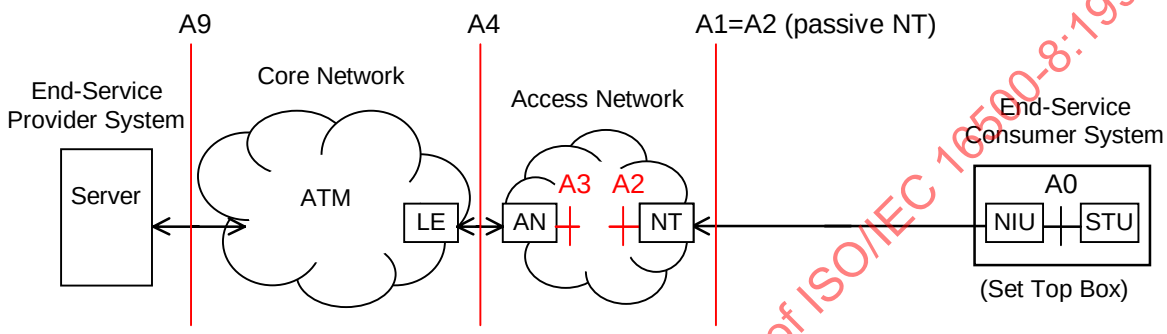


Figure 7-2 — Architecture of a Network Delivery System

Since the NT being considered in this specification is a passive NT, then the A2 reference point (medium and protocol) coincides with the A1 reference point. Depending on the technology, the A3 interface may be identical to A1. For instance, in the ADSL technology, A3 is identical to A1, however for HFC, A3 is different than A2 in terms of medium and lower layers protocol (fiber versus coax).

7.3 Access Network Management Architecture

The model presented here allows for defining an Access Network Management Information Base (MIB) which represents the Access Network resources that are visible to a Network Management System (NMS) in the S5 flow for managing the Access Network across the A8 management interface. The model supports all the management scenarios described in DAVIC 1.3.1a Part 4. That is, the model can be used by a NMS to manage the Access Network regardless of whether the NMS also manages other subsystems. In the management scenarios, Element Management Systems (EMS) may be introduced in conjunction to the NMSs. Figure 7-3 is based on management scenario 5 of DAVIC 1.3.1a Part 4. In this figure an EMS is introduced. The EMS may be provided in order to manage the components of the Access Network.

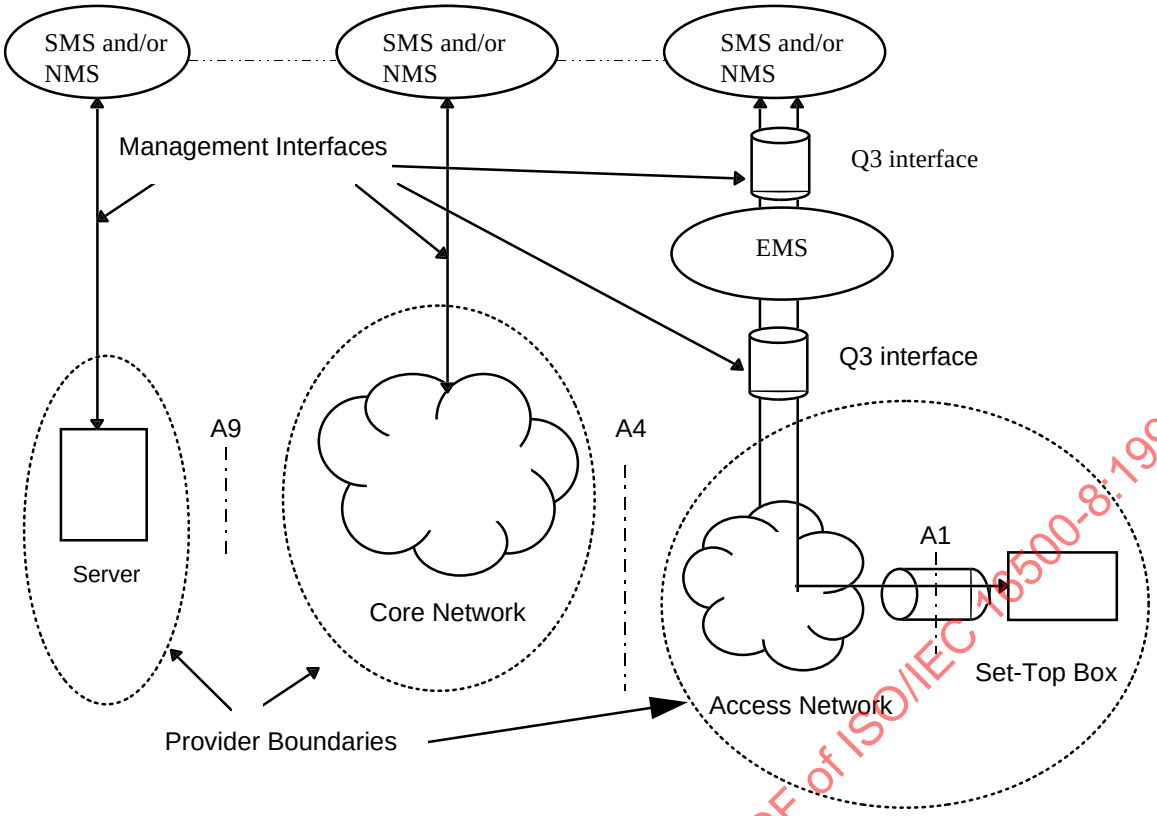


Figure 7-3 — Network Management Architecture - An Example

The Access Network resources that need to be managed include hardware and software resources. Resources that are of particular interest in the S5 flow are those needed for supporting the S1, S2, S3, and S4 flows. Figure 7-4 shows the DAVIC reference points, transaction flows and functional entities are relevant to the Access Network management interface. Details of these DAVIC components can be found, in particular, in ISO/IEC 16500-1 Figure 7.4-1 (Delivery System Control Plane Model) and Figure 8.0-1 (Functional Entities and Relationships).

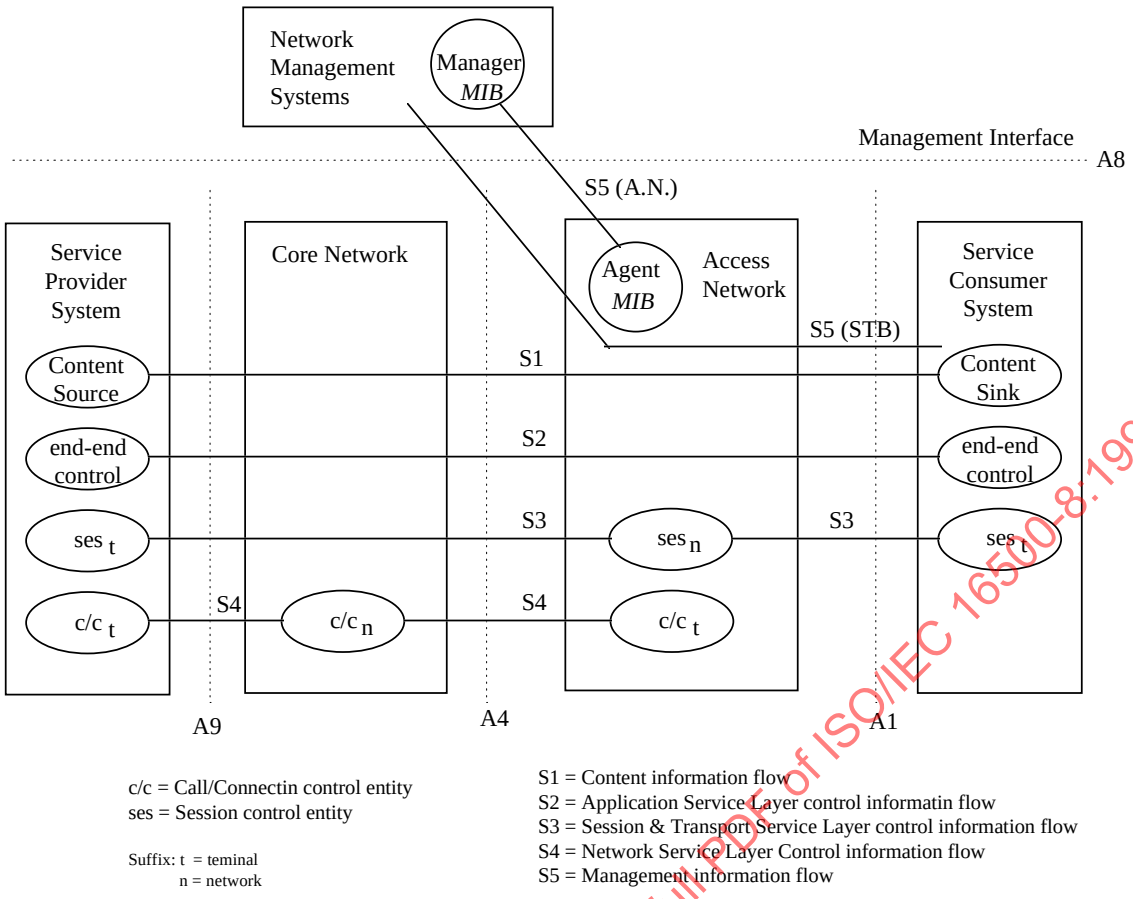


Figure 7-4 — Context of Access Network Management Interface

In Figure 7-4, the Session control entity ses_n and the Call/Connection control entity c/c_n are showed as located in the Access Network. This is one of the physical scenarios, namely Physical Scenario #1, shown in ISO/IEC 16500-2 subclause 9.2. It should be noted that this is not the only physical scenario supported by the model. The Access Network supports more functionality in this physical scenario then in other scenarios. Therefore, the model should be able to support the other physical scenarios also.

This section also models the access network resources that support interface initialization (i.e., the one using DIIP) of the Set-Top-Boxes (STB), the S3 and S5 flows. The S3 flow supports service related control (such as DSMCC UN Config, DSMCC UN Session, and Switched Digital Broadcast Channel Change). The S5 flows support the management of the subtending STBs and the Access Network itself.

The model presented here is a CMIP-based information model. It is an extension to the existing CMIP-based information models with DAVIC specific object classes defined in this document. Many managed object classes defined in ITU-T Recommendations, ATM Forum and industry specifications are reused.

7.4 Access Network Management Functions

Management of the Access Network through the A8 interface provides the functions that allow a manager to monitor and control the network resources. The functions are partitioned according to configuration management, fault management, performance management, accounting management, and security management.

S5 flow is related to network management. Only configuration, fault and performance management are addressed in this version of this part of ISO/IEC 16500. Management of the Core Network and the End-Service Provider System (Server) is logically separated from the management of the Access Network. However, as shown in the management scenarios in ISO/IEC 16500-1, in a physical implementation, the Access Network, Core Network,

and/or Server may be managed by the same system. It is assumed that accounting functions are performed outside the Access Network. Security management is for future study.

7.4.1 Configuration Management Functions

The features and functions that the Access Network Management should provide encompass static network representation as well as provisioning of network resources.

Following is a list of functions that should be implemented for configuration management:

- Functions that support service management system Work Orders (WO) to set up and provision or tear down equipment and end points within a communication circuit.
- Software version management: allow the control and monitoring of the software versions that are currently in place in the Network nodes.
- Dynamic software upgrade. Download of software release. An automatic switch-over is performed by the Network Nodes when the software download is complete.
- Memory backup and restoral. This involves backing up the components of the Access Networks and to store the configuration including ports parameters, cross connects etc.
- Node turn-up : Support the initial installation of a new node in the access network by offering the ability to download initial parameters that are needed. The download may be either a binary image or a series of messages to set up essential parameters.

7.4.2 Fault Management Functions

In the Access Network, the fault management should aid at detecting hardware or software problem in a non-ambiguous way. In order to achieve this, the essential functions that need to be provided by the Access Network may be summarized as follows:

- Management of alarm filters. The Access Network should support functions to filter events and notifications. 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 prior to the reporting. and/or logging of alarms
- Alarm Severity assignment: to associate a perceived severity to alarms, which may be generated by the Network Node. As an example, 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 should support requests to assign and modify a severity to each possible alarm.
- Alarm storage and retrieval through history logs.
- Support Loopback testing functions.

7.4.3 Performance Management Functions

Performance Management (PM) includes detection of Threshold Crossing Alerts and Trend Analysis. Its role is to help analyze the performance of the network and the Quality Of service provided by the network.

Some of the features associated with performance management in Access Network are listed below:

- PM data collection: the functions that support assignment of periodic PM data interval, suspend and resume the PM data collection, reset a PM data, and schedule the PM data collection.
- Storage of PM data: functions to support assignment of history duration, read (possibly using some filtering criteria, e.g. read only non-zero data) the PM data storage from the history logs, and delete the PM history data.
- Threshold monitoring: the functions that accept a PM threshold and report a threshold crossing.
- PM data reporting functions: This function refers to the ability of reporting PM data either on a scheduled basis or on an on demand basis. Functions to turn on or off the scheduled PM data report.

7.5 Access Network Management Protocol Stack

ISO/IEC 16500-5 (High and mid-layer protocols) recommends either CMIP ITU-T X.711, SNMP RFC 1157, or SNMPv2 RFC 1902 shall be used as a protocol suite to manage the Access Network.

Interworking of SNMP based network elements and CMIP based network elements may be supported as defined in Clause 6 (System Management Architecture) above.

The object model presented later in this section is described in GDMO (ITU-T X.722) and therefore implemented on a CMIP stack described in ISO/IEC 16500-5.

7.6 Management Information Model Overview

7.6.1 Managed Object Classes

In the access network information model, the scope of an Access Network includes an access node (e.g., such as a Host Digital Terminal - HDT) and its subtending distributed components (e.g., such as the Optical Network Units - ONU, Network Terminals - NT, and the fiber, cable or copper pair). It also includes the physical and logical resources within the access node and it subtending components. The resources considered include hardware, software, transport termination points (TPs) of the various transport layers (e.g., SDH or PDH, ATM, DAVIC Information Flow) at the A4 interface (i.e., with the code network) and A1 interfaces (i.e., with the CPEs), intra access network transport TPs, Cross-connections that support nail-up video services. In addition, it also includes logical resources that support the management of the access network.

This version of the specification provides the basis for a generic information model for the access network. In future specifications, extension to the model for specific access network architectures may be made, e.g., through subclassing. Table 7-1 identifies the object classes that form the basis of the generic information model. These object classes are either defined in this document or specified in other documents (international standards or industry). In the table, the object classes are also grouped into fragments for related functionalities.

Table 7-1 — Object classes in the Access Network Information Model

Fragment	Object Class	Source
Access Network	managed Element R1	ITU-T M.3100
Hardware	equipment R1	ITU-T M.3100
	equipment Holder	ITU-T M.3100
	circuit Pack	ITU-T M.3100
Software	software Unit	ITU-T X.744
	software Distributor	ITU-T X.744
Video Transport Specific Termination Points	multimedia TTP Bidirectional/Sink/Source	*Clause 7.7*
	service Related UN Channel CTP Bidirectional	*Clause 7.7*
	service Related UN Channel TTP Bidirectional	*Clause 7.7*
	access Network Management Channel CTP Bidirectional	*Clause 7.7*
	access Network Management Channel TTP Bidirectional	*Clause 7.7*
	cpe Management Channel CTP Bidirectional	*Clause 7.7*
	interface Initialization CTP Bidirectional	*Clause 7.7*
	interface Initialization TTP Bidirectional	*Clause 7.7*
SDH/SONET Specific Termination Points (Note 1)	Optical SPI TTP Bidirectional	ITU-T G.774
	Electrical SPI TTP Bidirectional	ITU-T G.774
	rs CTP Bidirectional	ITU-T G.774
	ms TTP Bidirectional	ITU-T G.774
	ms CTP Bidirectional	ITU-T G.774
	aug Bidirectional	ITU-T G.774
	au4 CTP Bidirectional	ITU-T G.774
	vc4 TTP Bidirectional	ITU-T G.774
	au4 4c CTP Bidirectional	Bellcore GR-1114
	vc4 4c TTP Bidirectional	Bellcore GR-1114
PDH Specific Termination Points	ds3 Line TTP Bidirectional	ANSI T1.240
	ds3 Line CTP Bidirectional	ANSI T1.240
	ds3 Path TTP Bidirectional	ANSI T1.240
	ds3 PLCP Path TTP Bidirectional	ATMF af-nm-0027
	ds3 PLCP Path CTP Bidirectional	ATMF af-nm-0027
ATM Specific Termination Points & Resources	tc Adaptor TTP Bidirectional	ATMF af-nm-0027
	vp CTP Bidirectional	ATMF af-nm-0027
	vp TTP Bidirectional	ATMF af-nm-0027
	vc CTP Bidirectional	ATMF af-nm-0027
	vc TTP Bidirectional	ATMF af-nm-0027
	interworking vc TTP Bidirectional (for non-ATM A1)	ATMF af-nm-0072
	aalProfile	ATMF af-nm-0071
	dss2 Sign Channel Tp	Bellcore GR-1114
	atm Saal Uni Protocol Profile	Bellcore GR-1114
	uni (or GR-1114: uniBCR)	ATMF af-nm-0027

Table 7-1 — Object classes in the Access Network Information Model (Cont.)

Fragment	Object Class	Source
Nail-up Connection	fabric R1	ITU-T M.3100
	mp Cross Connection	ITU-T M.3100
	cross Connection	ITU-T M.3100
Video Service Related	broadband Drop	*Clause 8*
	cpe	*Clause 8*
	cpe Authorization	*Clause 8*
	broadcast Program	*Clause 8*
	broadcast Program Feature	*Clause 8*
Generic Management Support	event Forwarding Discriminator	ITU-T X.721
	log	ITU-T X.721
	object Creation Record	ITU-T X.721
	object Deletion Record	ITU-T X.721
	attribute Value Change Record	ITU-T X.721
	state Change Record	ITU-T X.721
Fault Management	alarm Record	ITU-T X.721
	alarm Severity Assignment Profile	ITU-T M.3100
	current Alarm Summary Control	ITU-T Q.821
	testActionPerformer	ITU-T X.745
PDH Specific Termination Points	Generic Performance Monitoring Objects	ITU-T Q.822
	SDH Performance Monitoring Objects	ITU-T G.744-01 /06
	SONET Performance Monitoring Objects	ANSI T1.119-02
	PDH Performance Monitoring Objects	ANSI T1.247
	ATM Performance Monitoring Objects	ATMF af-nm-0027

Note 1*:*
These objects are needed if SDH or SONET is used between the core network and the access network.

7.6.2 Object Class Relationships

For video-specific object classes, their inheritant relationship and naming relationship are shown in Figure 7-5 and Figure 7-6 below. For object classes that are defined in other documents, the inheritant and naming relationship can be found in the corresponding source documents.

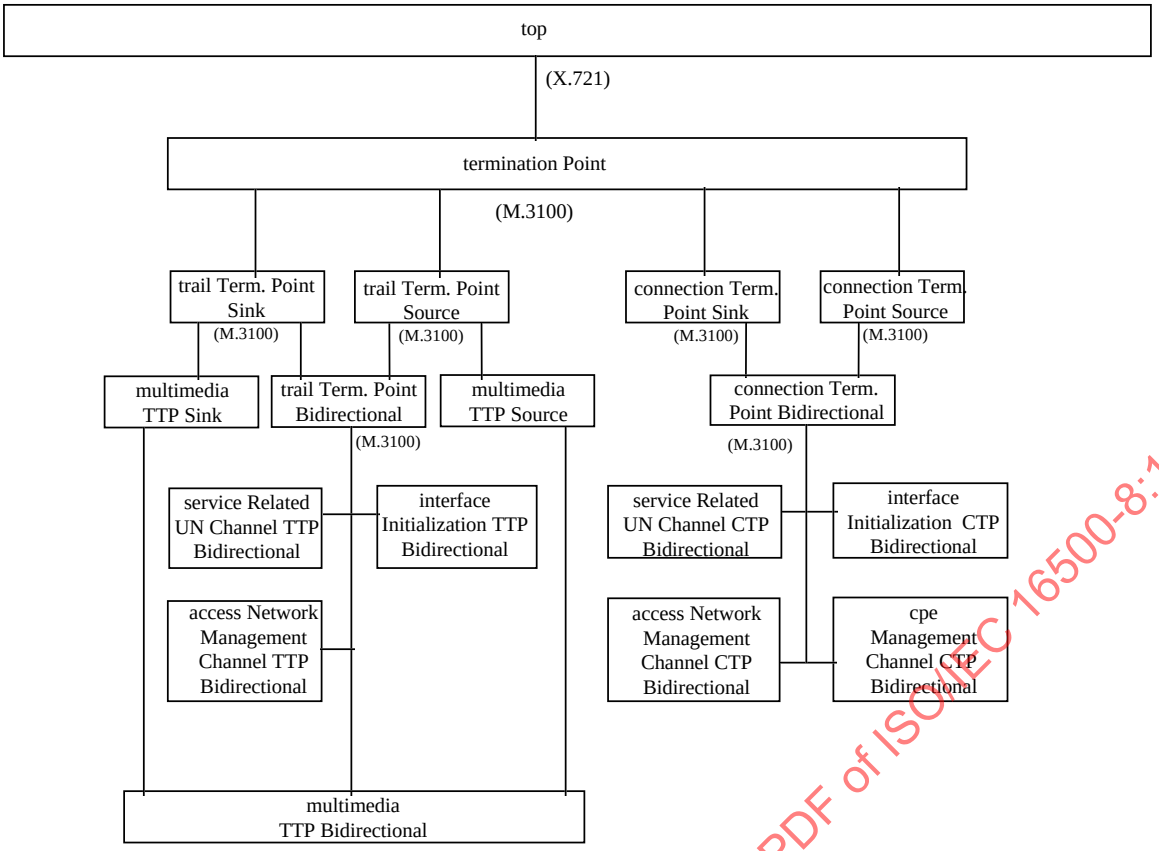


Figure 7-5 — Inheritance Relationship of Video-Specific Object Class

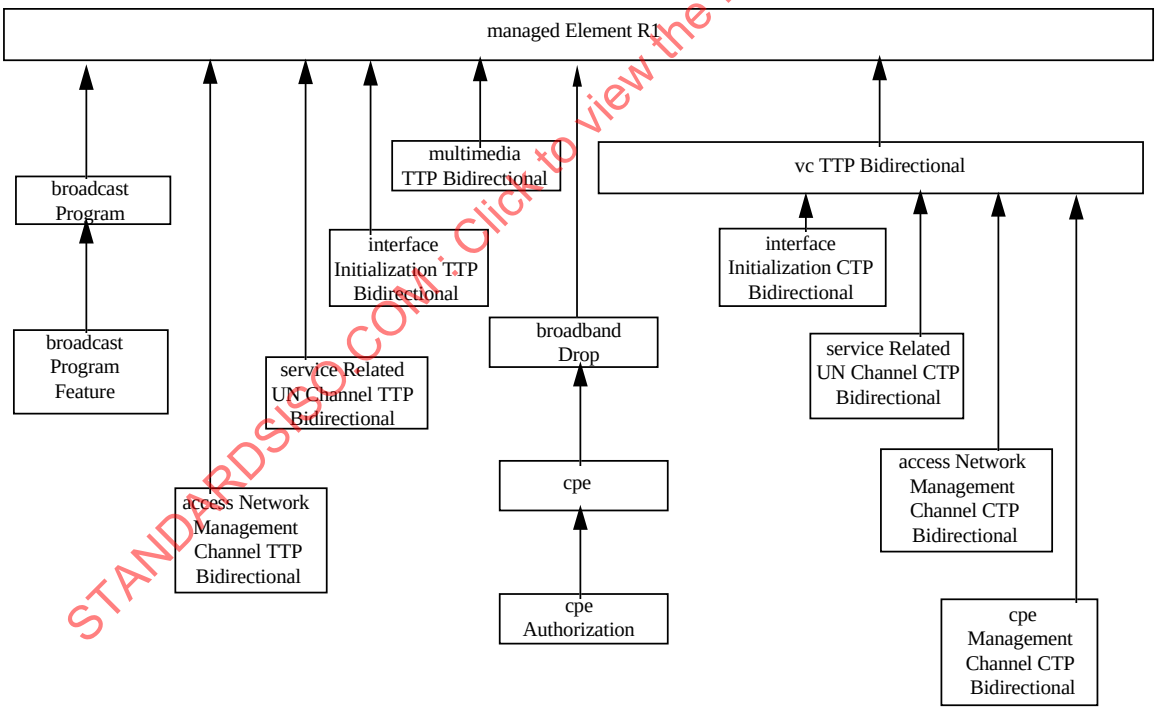


Figure **7-6** — **Naming Relationship of Video-Specific Object Class**

The following figure shows the containment relationship of the physical components of a access network. It also shows the cardinality of each association. The local root of the access network is represented by a

managedElementR1 object, which contains an equipmentR1 object representing the access node. The access node in turn contains multiple ONUs (represented by the equipmentR1 objects). Each ONU in turn contains multiple Network Terminals (NT), which are also represented by instances of the equipmentR1 object classes. Within the access node, ONUs and NTs, the shelf (or slot) and plug-in card are represented as instances of euipmentHolder and circuitPack.

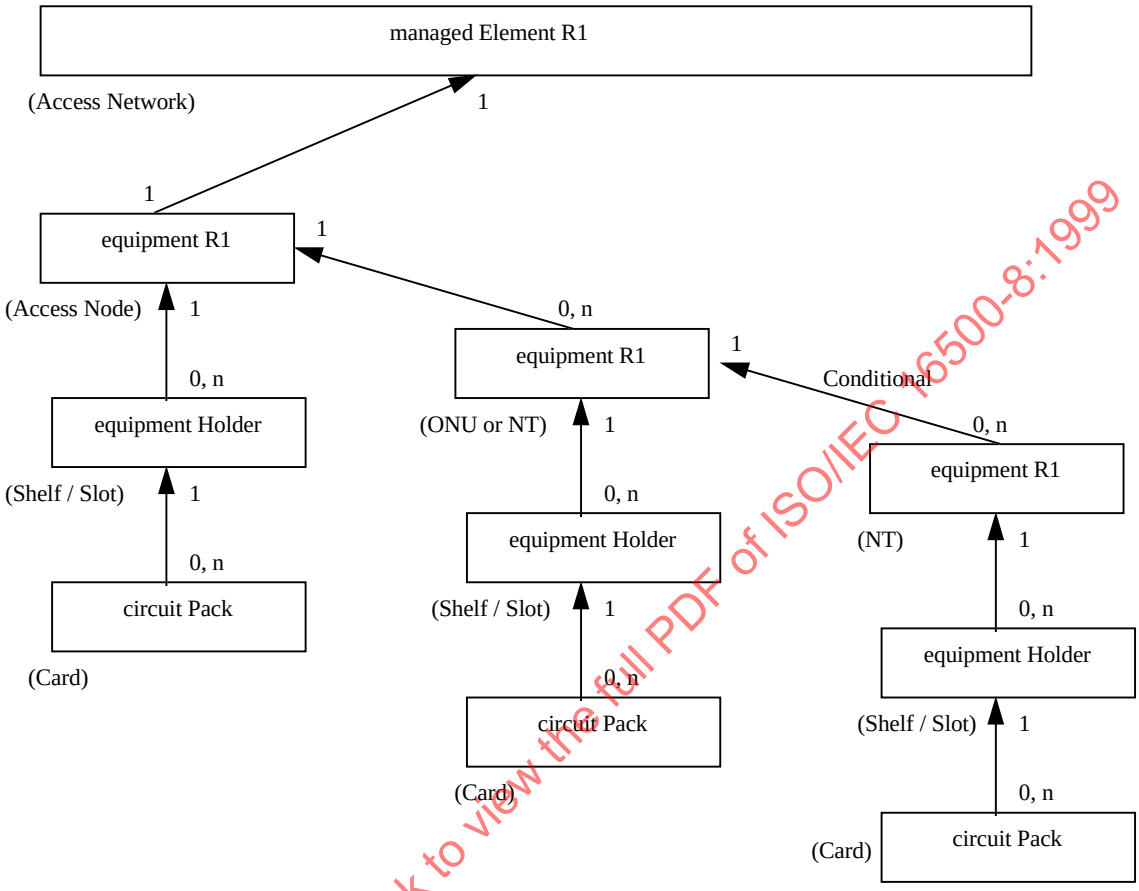


Figure 7-7 — Example of Physical Components Entity Relationship.

The following figure shows the relationship among the termination points. Only the video-specific termination points and their directly associated termination points are shown. Therefore the SDH/SONET and PDH termination points are not shown in the figure. In general, TTP (trail termination point) objects are contained directly in the Managed Element. TTP and CTP (connection termination point) are associated with each other through the downstreamConnectivityPointer and upstreamConnectivityPointer. A trail is terminated/originated with two TTP objects. An instance of tcAdaptorTTP at the A1 interface (i.e., toward the CPE) is supported by an instance of multimediaTTP object. An instance of tcAdaptorTTP at the A4 interface (i.e., toward the core network) is supported by an instance of vc4TTP or other SDH/SONET object.

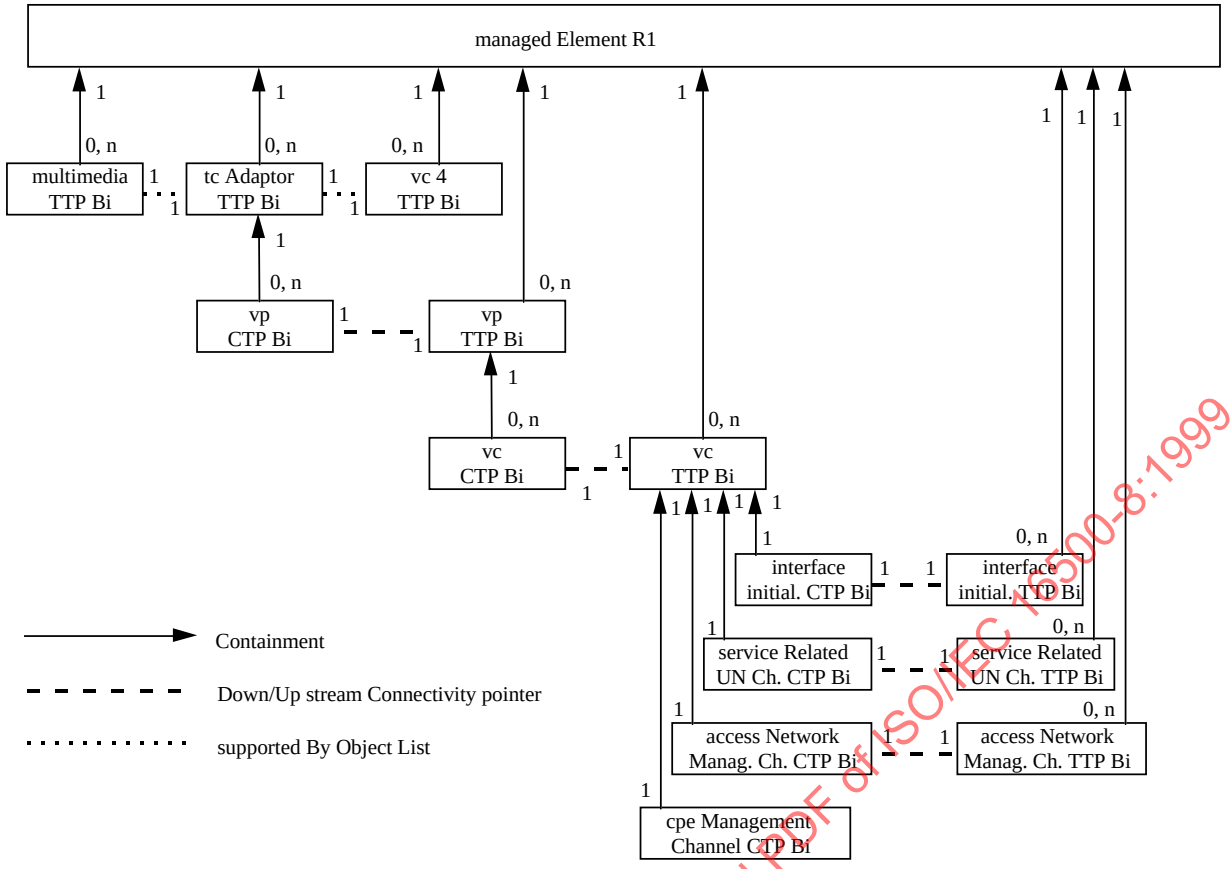


Figure 7-8 — Entity Relationship of Termination Points

The following figure shows the relationship between transport termination point objects and their underlying hardware resources. In general, TTP objects are supported by hardware objects, such as circuitPack.

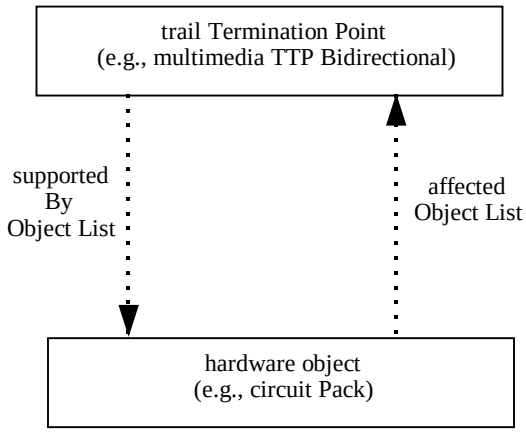


Figure 7-9 — Entity Relationship of TTP and Hardware

The following figure shows the relationship of the fabric, mpCross-connection, cross-connection, and termination point objects. In the Point-to-Multipoint cross connect, transport information is sent from a termination point sink object to one or more termination point source objects. It should be noted that the sink and source characteristic of a termination point is from the perspective with respect to the exterior of the network element (or access network in this case). That is, a termination point sink object is where the transport information from the network terminated. On the other hand, the termination point source object is where the transport information to the network originated. Inside of the network element (or the access network in this case), transport information received from the network at the termination point sink is cross connected to a termination point source and transmitted to the network.

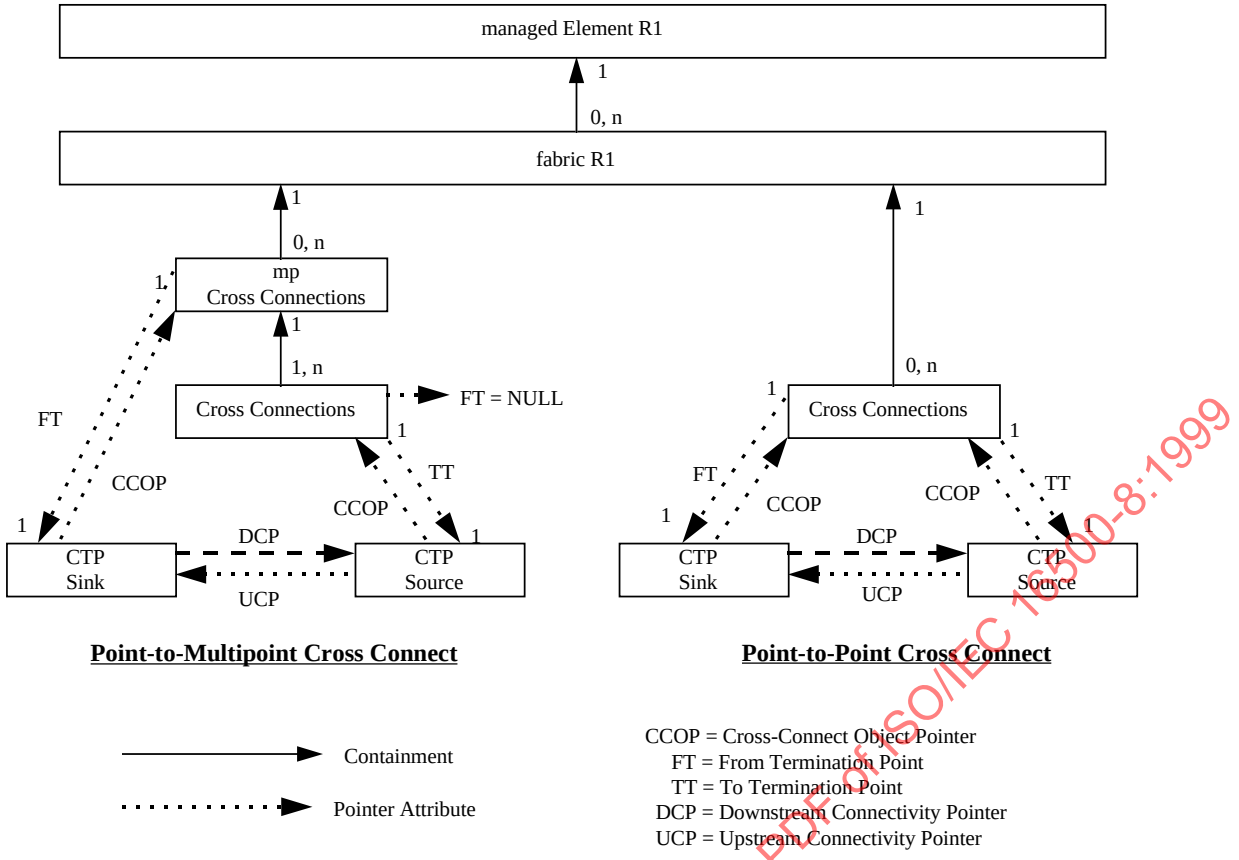


Figure 7-10 — Entity Relationship of Cross-Connect

7.6.3 Application of the Model to DAVIC Information Flows

The transport model of the Access Network is based on the generic functional architecture of transport networks described in ITU-T G.805 and the generic network information model defined in ITU-T M.3100. Two major types of object classes are used for the transport function of the Access Network, namely Trail Termination Point (TTP) object classes and Connection Termination Point (CTP) object classes. A trail is an entity in a transport layer network, which is responsible for the integrity of transfer of information from one or more other layer networks. A connection is an entity, which is responsible for the transparent transfer of information between its two end points, namely the CTPs. A connection is a component of a trail. Several connections can be bundled into a higher rate trail. A sequence of one or more connections are linked together to form a trail. A connection may be either uni- or bi- directional. A trail is composed of two TTPs and one or more connections and associated CTPs.

The Access Network system management information model consists of object classes representing the transport configuration, hardware and software resources of the Access Network. It also includes object classes to support system management of the Access Network. A managing system manages the resources of the Access Network through managing instances of these object classes.

The following subsection describes the information model of an Access Network for supporting the transport configuration of information flowing through the Access Network. In the information model, object instances are used to represent resources or functions of the Access Network. These instances are related to each other through containment and optionally through referencing. Figures are used to illustrate the relationship. In the figures in this section, object instances are illustrated as boxes. A containment relationship is depicted pictorially as a box containing other box. A referencing relationship is shown using arrows pointing from the referencing object to the referenced object.

7.6.3.1 Initial Transport Configuration

Figure 7-11 shows the initial transport configuration of an Access Network. The Access Network is modeled as an instance of the managedElementR1 object class and represented by the outermost box in the figure. There are several factors that determine the transport configuration of an Access Network: whether ATM is employed at the A1 interface (i.e., at the consumer side of the access network) for carrying the DAVIC information flows (e.g., S1, S2, S3, S4, S5 and the DIIP flows), whether PDH or SONET/SDH is used at the A4 interface (i.e., at the core network side of the access network) for the physical transport of the information flows, and whether the Session control entity (ses_n) and Call/Connection control entity (C/C_n) are located in the Access Network. In the figure, both SDH/SONET and PDH are shown in the same figure, instead of using several figures, with the understanding that it is not required for an Access Network implementation to support both transport technologies at the same time.

At the A4 interface, if SDH/SONET technology is used for the physical transport, then the SDH termination point objects (namely optical SPI TTP Bidirectional or electrical SPI TTP Bidirectional, rs CTP Bidirectional, rs TTP Bidirectional, ms CTP Bidirectional, ms CTP Bidirectional, aug Bidirectional, au4 CTP Bidirectional or au44c CTP Bidirectional, vc4 TTP Bidirectional or vc44c TTP Bidirectional) will exist.

If PDH technology is used for the physical transport at the A4 interface, the PDH termination point objects will exist in the Access Network. In the figure, PDH objects for the case of DS3 are used (namely, ds3 Line TTP Bidirectional, ds3 CTP Bidirectional, ds3 Path TTP Bidirectional, ds3 PLCP Path CTP Bidirectional, and ds3 PLCL Path TTP Bidirectional).

It is assumed that in both the SDH/SONET and PDH cases, ATM is used for carrying the S1, S2, S3, S4, and S5 flows. The ATM termination point object (namely, tc Adaptor TTP Bidirectional, vp CTP Bidirectional, vp TTP Bidirectional, vc CTP Bidirectional, and vc TTP Bidirectional -- or it subclass interworking VC TTP Bidirectional) will exist at the A4 interface.

At the A1 interface, the access lines (e.g., copper pair, coax cable, or fiber) connect the service consumers with the Access Network to support the physical transport of information flows. In the Access Network, the origination and termination function of an access line is modeled as a multimedia TTP Bidirectional (or an-multimedia TTP Source for broadcast only access network) instance. It is also assumed that the A1 interface is ATM-based.

For simplicity, only one A4 side vp TTP Bidirectional object and one A1 side vp TTP Bidirectional object are shown in the figure and all the vc CTP Bidirectional objects are contained in these two objects. In reality, there may be more than one vp TTP Bidirectional objects on both sides.

In the initial stage, for each access line at the A1 interface, there shall be one DAVIC Interface Initialization connection (STB interface initialization using the DAVIC Interface Initialization Protocol, DIIP, for establishing S3), multiple S5 connections (one for each STB). For each access line, the DIIP connection is shared by all its service consumers. At the A4 interface, there shall be one S5 connection for the Access Network itself and multiple S5 connections (one for each STB). The CTPs of these connections are modeled as interface Initialization CTP Bidirectional, cpe Management Channel CTP Bidirectional, and access Network Management Channel CTP Bidirectional.

In the Access Network, the function of formulating/originating and terminating/processing the DIIP messages is modeled as instances of interface Initialization TTP Bidirectional. An interface Initialization TTP Bidirectional object and an interface Initialization CTP Bidirectional object are associated with each other using the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects. In the figure, only the downstream Connectivity Pointers are shown.

The function of formulating/originating and terminating/processing the S5 flow for managing the Access Network is modeled as instances of access Network Management Channel TTP Bidirectional. A access Network Management Channel TTP Bidirectional object and an access Network Management Channel CTP Bidirectional object are associated with each other using the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

The Access Network is responsible only for the transparent transfer of the S5 flows that belong to the STBs. Therefore the TTPs of these S5 flows do not exist in the Access Network. The Access Network is responsible for connecting the A4 side cpe Management Channel CTP Bidirectional with the A1 side cpe Management Channel CTP Bidirectional for completing the end-to-end STB S5 flows. The connecting relationship is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

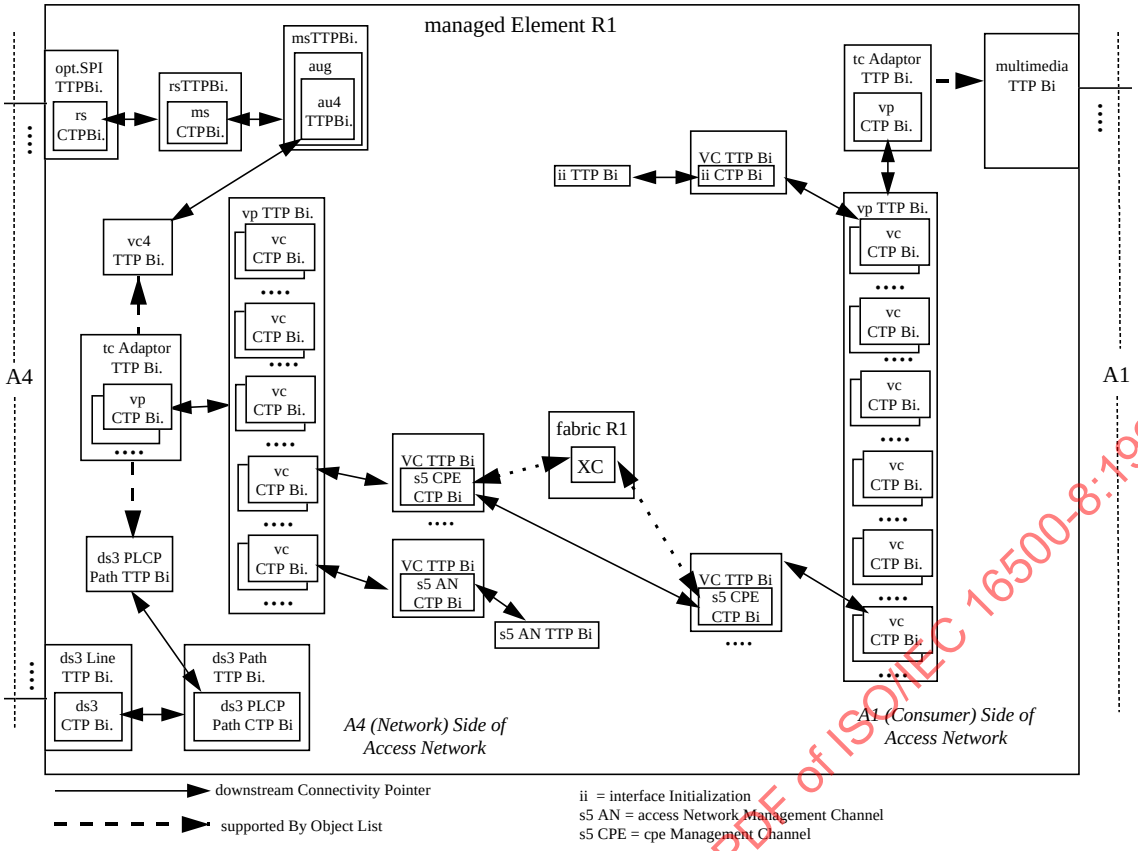


Figure 7-11 — Access Network Initial Transport Configurations

7.6.3.2 Transport Configuration for S3 Flow

Figure 7-12 shows the configuration of the Access Network when the S3 connections have been established. The S3 connection is set up using DIIP.

The CTP of the S3 connections are modeled as service Related UN Channel CTP Bidirectional. In the figure, it is assumed that the Session control entity ses_n is not located in the Access Network. If ses_n is located in the Core Network, the scenario is described in Physical Scenario #2 (where the Call/Connection control entity C/C_t is located in the STB) and Physical Scenario #3 (where C/C_t is located in the core network) of ISO/IEC 16500-2. In both cases, the Access Network is responsible for transparent transfer of the S3 flow, i.e., directly connecting the A4 side and A1 side CTPs with each other.

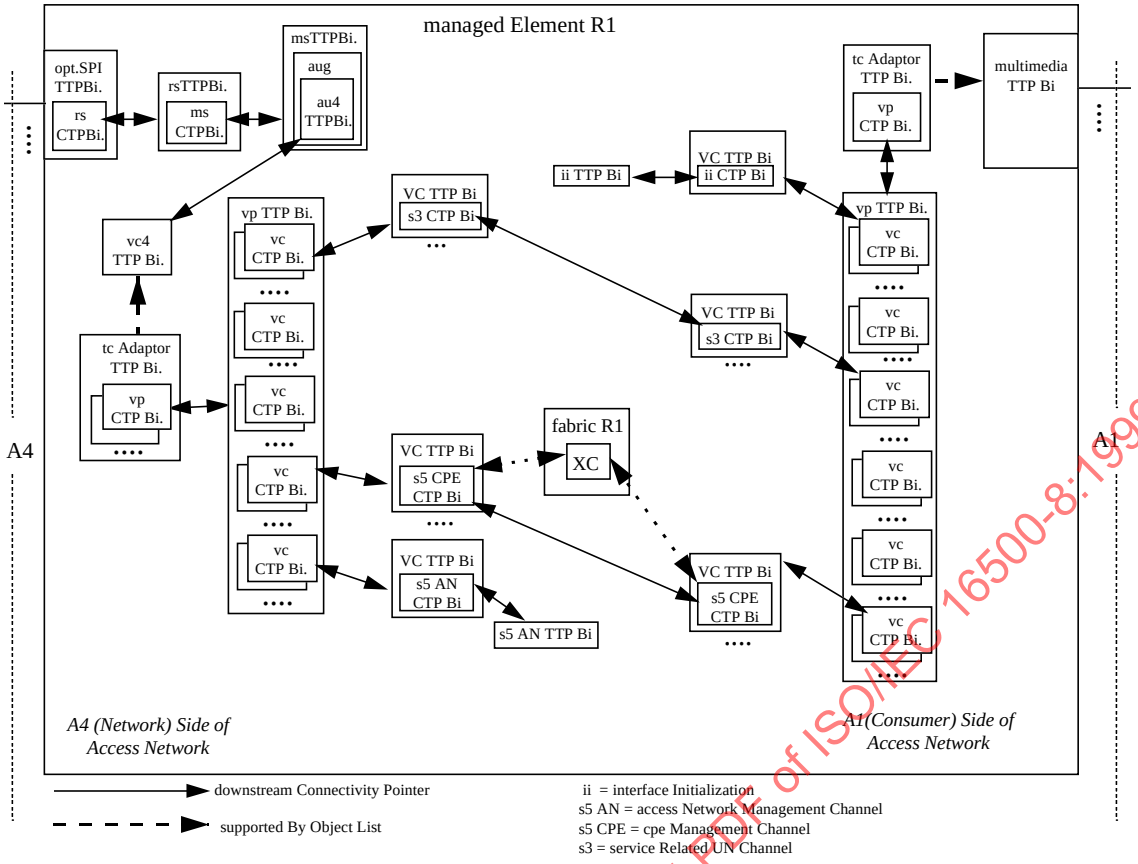


Figure 7-12 — S3 Flow, ses_n not in the Access Network

Figure 7-13 shows the configuration of the Access Network when the Session control entity is located in the Access Network. This is scenario is described in Physical Scenario #1 of ISO/IEC 16500-2. The termination and origination functions of the Session control entity is modeled as the service Related UN Channel TTP Bidirectional object. The service Related UN Channel CTP Bidirectional and the service Related UN Channel TTP Bidirectional are associated with each other using the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

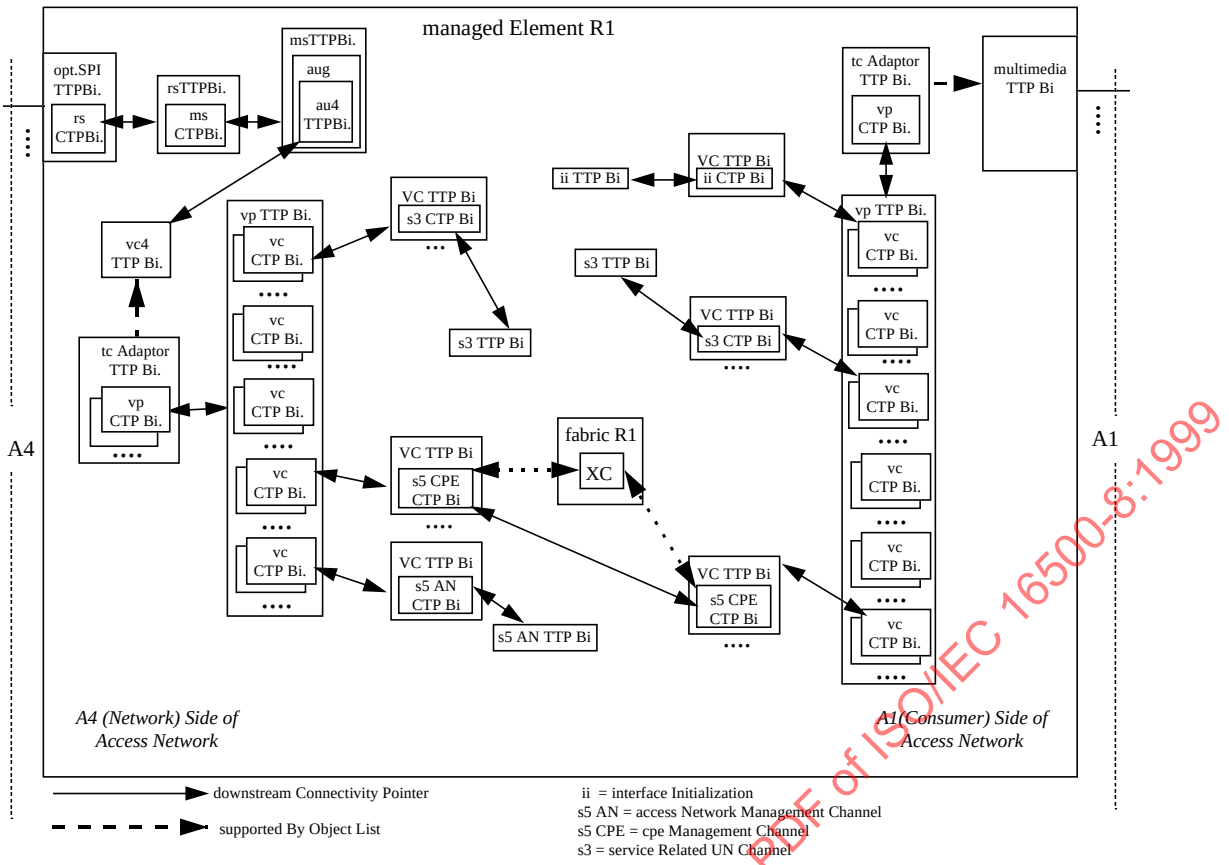


Figure 7-13 — S3 Flow, ses$_n$ not in the Access Network

7.6.3.3 Transport Configuration for HFC-Based A1 Interface

In the above two subsections, it is assumed that the A1 interface is ATM-based. This subsection describes the configuration when the A1 interface is not ATM-based, e.g., HFC. At the A1 side of the access network, the multimedia TTP Bidirectional will directly contains the interface Initialization CTP Bidirectional, service Related UN Channel CTP Bidirectional and cpe Management Channel CTP Bidirectional objects. At the A4 side of the access network, the service Related UN Channel CTP Bidirectional and cpe Management Channel CTP Bidirectional objects will contained in the interwork VC TTP Bidirectional object. The following figure shows the configuration when the Session control entity is not located in the access network. If the Session control entity is located in the access network, then the service Related UN Channel TTP Bidirectional will exist in the access network and the service Related UN Channel CTP Bidirectional will not be connected to which other. Also the A4 side service Related UN Channel CTP Bidirectional will be contained in the vc TTP Bidirectional object instead of the interwork VC TTP Bidirectional object.

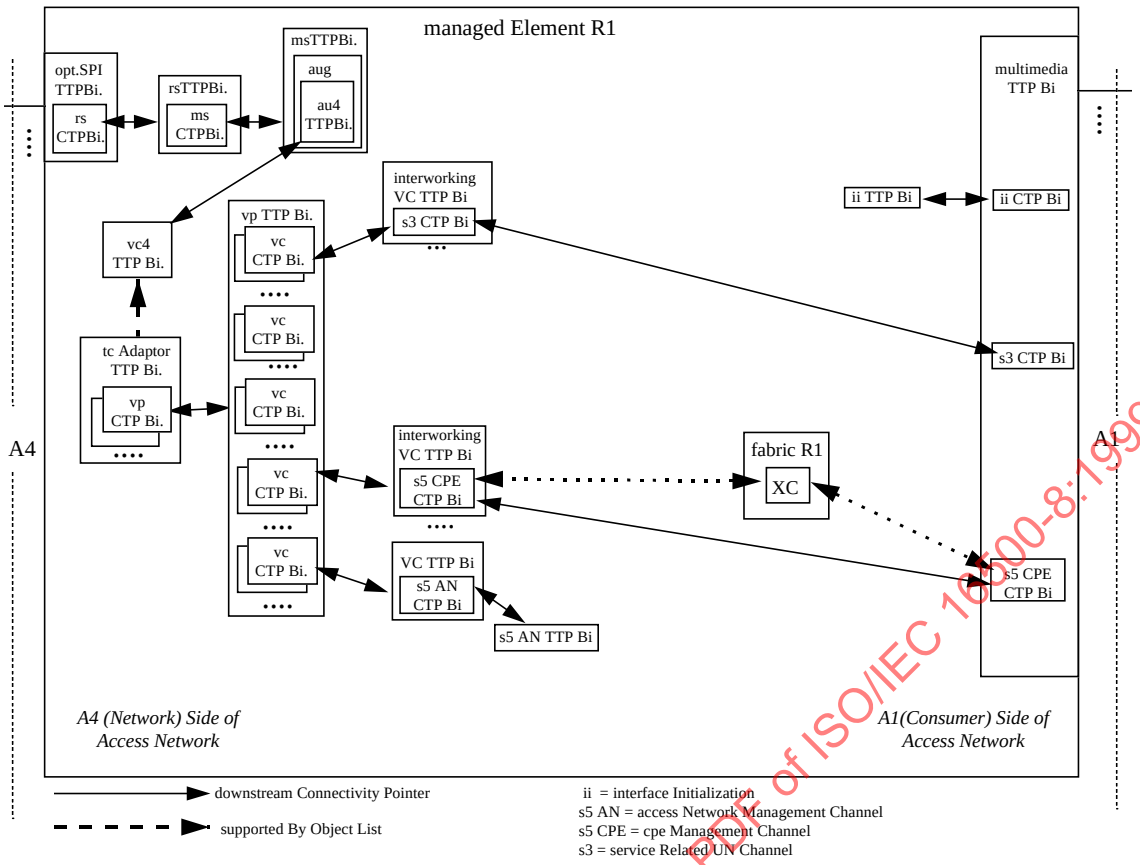


Figure 7-14 — Transport Configuration for HFC-Based A1 Interface

7.6.3.4 Transport Configuration of S1, S2, and S4 Flows

The S1 and the S2 flows are information flows from the service provider to the service consumer. The S1 flow is the content information and is uni-directional while the S2 flow is the application service layer control information and is bidirectional. The S1 and S2 flows are carried through the access network by the underlying transport entities (such as ATM, SDH/SONET or PDH and are therefore transparent to the access network and invisible in the A8 management interface to the access network. The S4 connection is set up by using the ATM Access Signaling Channel (VPI/VCI=0/5). The ATM Access Signaling Channel can be managed using the dss2 Sign Channel Tp and dss2 Sign Point objects defined in Bellcore GR-1114-CORE.

7.6.3.5 Transport Configuration of Nailed-Up Service

In some applications, the video services could be semi-permanently set-up through transport configuration instead of signaling. That is, even though the S1 and S2 are still transparent to the access network, the video services could still be set-up through cross-connecting the underlying transport resource such as ATM virtual path (VP) or virtual channel (VC). Figure 7-15 shows the configuration of cross-connect at the ATM VP and VC level to support video services. In these cases, similar to the S1 and S2 flows, the S3 and Interface Initialization information flows (if any) are carried as ATM payload and not processed by the Access Network, and thus invisible in the Access Network A8 interface.

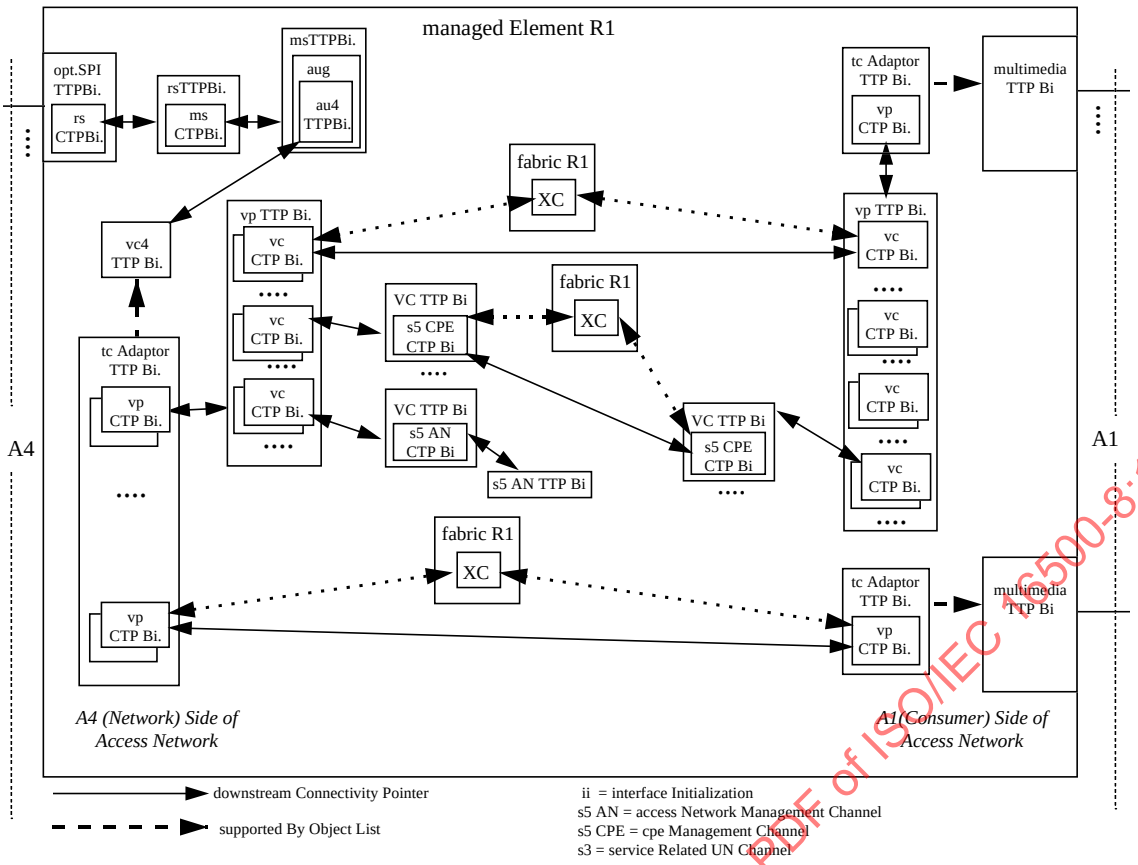


Figure 7-15 — Nailed-Up Services Supported by ATM Cross-Connect

7.6.3.6 Hardware and Software Resources and Support Objects

The following figure illustrates some of the transport, hardware, software, and management support objects and their relationship. For simplicity, not all the transport termination points, hardware, software and support objects are shown. For example, performance monitoring objects are not shown in the figure. The connectivity pointers are also not shown.

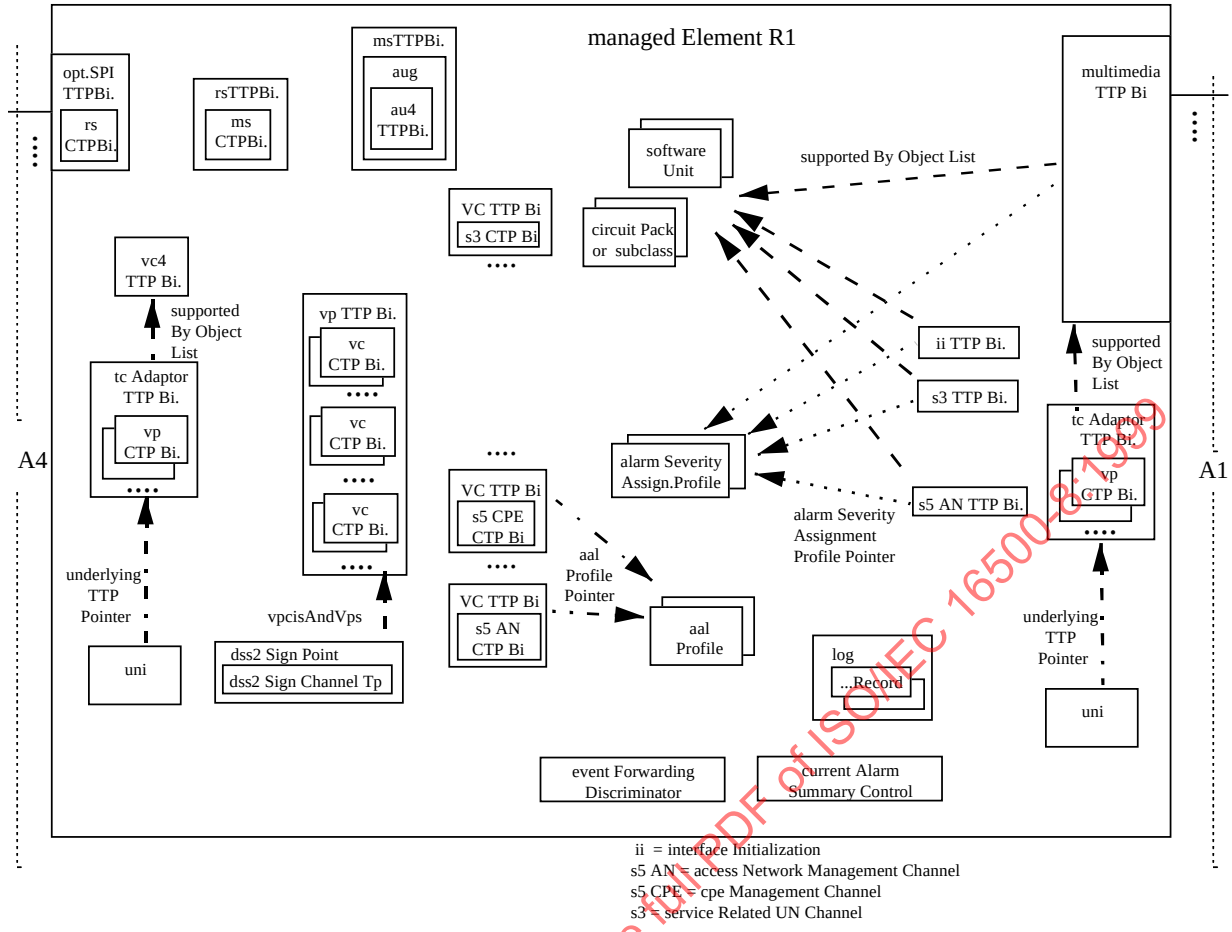


Figure 7-16 — Hardware, Software and Management Support Objects

7.6.4 Managed Object Class Overview

This section provides a protocol-neutral description of the video-specific managed object classes, which are defined formally in Clause 7 of this specification.

7.6.4.1 Multimedia TTP Bidirectional / Sink / Source

An instance of this object class represents a termination point of a multimedia trail. Each port of the access lines at the Access Network is represented by an instance of this object class.

- Attributes:

  ◊ multimedia TTP Id:

  This read-only attribute provides a unique name for an instance of the multimedia TTP Bidirectional object class within an Access Network (represented by an instance of the managed Element R1 object class

  ◊ access Topology:

  This read-only attribute indicates the access topology of the access line, e.g., ADSL, VDSL, FTTC (type 1, 2, etc.), HFC (550 MHz, 750 MHz, 1 GHz, etc.), FTTH, etc.

  ◊ access Media:

  This read-only attribute indicates the media type of the access line, e.g., copper, coax cable, fiber, etc..

  ◊ modulation Type:

This read/write attribute specifies the modulation type of the access line, e.g., QAM16, QAM64, QAM256, VSB16, QPSK, BPSK, CAP4, CAP16, etc.

◊ supported By Object List:

This read/write attribute identifies the supporting hardware resources by pointing to the corresponding equipment or circuit Pack objects.

◊ operational State

This read-only attribute indicates the operability of the access line, i.e., whether it is capable of performing its normal functions.

◊ administrative State

This read/write attribute is used to inhibit (lock) and allow (unlock) the flow of information through the access line.

◊ current Problem List

This read-only attribute is used to indicates the current existing problems, with severity, associated with the access line.

◊ alarm Severity Assignment Profile Pointer

This read/write attribute provides a pointer relationship to an alarm Severity Assignment Profile object.

◊ associatedObjectList

Identifies the associated VB51 interface or IDLC interface group.

◊ transmissionPathName

Name of the transmission path.

◊ physicalBitRate

kbit/s.

- Actions:

◊ operateLoopback

Instructs the access network to perform a loopback test on the access line associated with this termination point.

◊ releaseLoopback

Instructs the access network to release a loopback test on the access line associated with this termination point.

- Notifications:

◊ communications Alarm

This message is used to notify the managing system when a failure has been detected or cleared. The message shall include at least the following information:

* The nature of the alarm (i.e., communications alarm)
* The cause of the failure (e.g., loss of signal)
* The ID of the object reporting the alarm
* The severity of the failure (critical, major, minor, warning, indeterminate, or clear)
* The time and date failure was detected

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

◊ state Change

This notification is used to report value changes to the operational State and administrative State attributes.

- Relationship:

◊ One or more instances of this object class are contained in the managed Element R1 object which represent an Access Network.

◊ Each instance of this object class is referenced by an instance of the tc Adaptor TTP Bidirectional object class using the supported By Object List attribute.

◊ Each instance of this object class is associated with one or more supporting objects (e.g., the equipment Holder or circuit Pack objects). The association is indicated by the supported By Object List attribute of this object and the affected By Object List attribute of the supporting objects.

◊ Each instance of this object class is associated with an instance of the alarm Severity Assignment Profile object class. The association is indicated by the alarm Severity Assignment Profile Pointer attribute of the object.

7.6.4.2 Interface Initialization CTP Bidirectional

An instance of this object class represents a bidirectional termination point of the DAVIC Interface Initialization channel of an access line. The Interface Initialization channel is shared by all the Service Consumers of the access line for requesting S3 channel assignments. For ATM-based CPEs, this channel is operated over a predefined VPI/VCI and used the DAVIC Interface Initialization Protocol (DIIP) for the request messages. For non-ATM-based CPEs, this channel is operated using the MAC layer 2 protocol for the request messages.

- Attributes:

◊ interface Initialization CTP Id:

This read-only attribute provides a unique name for an instance of the interface Initialization CTP Bidirectional object class within its containing object.

◊ frequency:

For HFC architecture, this read/write attribute indicates the frequency (in KHz) used in carrier modulation. For analog signals, this value indicates the carrier frequency. For digital signals that have been modulated onto an Radio Frequency (RF) carrier, this attribute indicates the center frequency.

◊ frequency Spacing:

For HFC architecture, this read/write attribute indicates the size of the frequency slot in KHz for an RF signal.

- Actions:

None

- Notifications:

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

- Relationship:

- ◊ Instances of this object class exist at the A1 interface of the Access Network. For each multimedia TTP Bidirectional object there is one instance of interface Initialization CTP Bidirectional.
- ◊ Each instance of this object class is associated with one instance of the interface Initialization TTP Bidirectional object class. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

7.6.4.3 Interface Initialization TTP Bidirectional

An instance of this object class represents a termination point at which the DAVIC Interface Initialization information is originated/formulated and terminated/processed.

- Attributes:
  - ◊ interface Initialization TTP Id:

    This read-only attribute provides a unique name for an instance of the interface Initialization TTP Bidirectional object class within an Access Network (represented by an instance of the managed Element R1 object class)
  - ◊ supported By Object List:

    This read/write attribute identifies the supporting hardware and software resources by pointing to the corresponding resource objects, such as circuit Pack or software Unit.
  - ◊ operational State

    This read-only attribute indicates the operability of the Interface Initialization channel, i.e., whether it is capable of performing its normal functions.
  - ◊ administrative State

    This read/write attribute is used to inhibit (lock) and allow (unlock) the flow of information through the channel.
  - ◊ current Problem List

    This read-only attribute is used to indicates the current existing problems, with severity, associated with the channel.
  - ◊ alarm Severity Assignment Profile Pointer

    This read/write attribute provides a pointer relationship to an alarm Severity Assignment Profile object.
- Actions:

None

- Notifications:
  - ◊ communications Alarm

    This message is used to notify the managing system when a failure has been detected or cleared. The message shall include at least the following information:
    - * The nature of the alarm (i.e., communications alarm)
    - * The cause of the failure (e.g., loss of signal)
    - * The ID of the object reporting the alarm
    - * The severity of the failure (critical, major, minor, warning, indeterminate, or clear)
    - * The time and date failure was detected
  - ◊ attribute Value Change

    This notification is used to report value changes to the attributes of the object instance.
  - ◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

◊ state Change

This notification is used to report value changes to the operational State and administrative State attributes.

- Relationship:

◊ Instances of this object class exist at the consumer side of the Access Network. One or more instances of this object class are contained in the managed Element R1 object which represent the Access Network.

◊ Each instance of this object class is associated with an instance of the interface Initialization CTP Bidirectional object class. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

◊ Each instance of this object class is associated with one or more supporting objects (e.g., equipment Holder, circuit Pack, and software Unit). The association is indicated by the supported By Object List attribute of this object and the affected By Object List attribute of the supporting objects.

◊ Each instance of this object class is associated with an instance of the alarm Severity Assignment Profile object class. The association is indicated by the alarm Severity Assignment Profile Pointer attribute of the this object.

7.6.4.4 Service Related UN Channel CTP Bidirectional

An instance of this object class represents a termination point of a S3 connection. The S3 connection is used for session control.

- Attributes:

◊ service Related UN Channel CTP Id:

This read-only attribute provides a unique name for an instance of the service Related UN Channel CTP Bidirectional object class within its containing object.

◊ frequency:

For HFC architecture, this read/write attribute indicates the frequency (in KHz) used in carrier modulation. For analog signals, this value indicates the carrier frequency. For digital signals that have been modulated onto an Radio Frequency (RF) carrier, this attribute indicates the center frequency.

◊ frequency Spacing:

For HFC architecture, this read/write attribute indicates the size of the frequency slot in KHz for an RF signal. Typical spacings for video services are 6MHz and 8 Mhz.

◊ channel Type:

This read-only attribute indicates the type of the S3 flows. Possible channel types include DSMCC Channel Change Channel, UN Signaling Channel, and UN Configuration Channel.

◊ associated CPE:

This read-only attribute identifies the associated CPE of the S3 flow

- Actions:

None

- Notifications:

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

- Relationship:

◊ At the consumer side of the Access Network, zero or more instances of this object class may be contained in a vc TTP Bidirectional object.

◊ At the network side of the Access Network, zero or more instances of this object class may be contained in an vc TTP Bidirectional object.

◊ If the Session control entity is not located in the Access Network, then an instance of this object class at the consumer side and an instance of this object class at the network side are associated with each other. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

◊ If the Session control entity is located in the Access Network, then each instance of this object class is associated with an instance of the service Related UN Channel TTP Bidirectional object class, which is also contained in the Access Network. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

7.6.4.5 Service Related UN Channel TTP Bidirectional

An instance of this object class represents a termination point at which the S3 information is originated/formulated and terminated/processed. Instances of this object class exist in an Access Network if the Session control function is located in the Access Network.

- Attributes:

◊ service Related UN Channel TTP Id:

This read-only attribute provides a unique name for an instance of the service Related UN Channel TTP Bidirectional object class within an Access Network (represented by an instance of the managed Element R1 object class)

◊ supported By Object List:

This read/write attribute identifies the supporting resources by pointing to the corresponding equipment, circuit Pack, or software Unit objects.

◊ operational State

This read-only attribute indicates the operability of the S3 flow, i.e., whether it is capable of performing its normal functions.

◊ administrative State

This read/write attribute is used to inhibit (lock) and allow (unlock) the flow of information through the channel.

◊ current Problem List

This read-only attribute is used to indicates the current existing problems, with severity, associated with the channel.

◊ alarm Severity Assignment Profile Pointer

This read/write attribute provides a pointer relationship to an alarm Severity Assignment Profile object.

- Actions:

None

- Notifications:

◊ communications Alarm

This message is used to notify the managing system when a failure has been detected or cleared. The message shall include at least the following information:

- The nature of the alarm (i.e., communications alarm)
- The cause of the failure (e.g., loss of signal)
- The ID of the object reporting the alarm
- The severity of the failure (critical, major, minor, warning, indeterminate, or clear)
- The time and date failure was detected

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

◊ state Change

This notification is used to report value changes to the operational State and administrative State attributes.

- Relationship:

◊ If the network side Session control function is located in the Access Network, then zero or more instances of this object class may be contained in the Access Network.

◊ Each instance of this object class is associated with an instance of the service Related UN Channel CTP Bidirectional object class. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

◊ Each instance of this object class is associated with one or more supporting objects (e.g., equipment Holder, circuit Pack, and software Unit). The association is indicated by the supported By Object List attribute of this object and the affected By Object List attribute of the supporting objects.

◊ Each instance of this object class is associated with an instance of the alarm Severity Assignment Profile object class. The association is indicated by the alarm Severity Assignment Profile Pointer attribute of the this object.

7.6.4.6 Access Network Management Channel CTP Bidirectional

An instance of this object class represents a termination point of a S5 connection which is used for managing the Access Network.

- Attributes:

◊ access Network Management Channel CTP Id:

This read-only attribute provides a unique name for an instance of the access Network Management Channel CTP Bidirectional object class within its containing object.

- Actions:

None

- Notifications:

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

- Relationship:

◊ One instance of this object class exists in the Access Network and is contained in an vc TTP Bidirectional object at the network side of the Access Network.

◊ An instance of this object class is associated with an access Network Management Channel TTP Bidirectional object. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

7.6.4.7 Access Network Management Channel TTP Bidirectional

An instance of this object class represents a termination point at which the Access Network S5 information is originated/formulated and terminated/processed. One instance of this object class exists in an Access Network.

- Attributes:

◊ access Network Management Channel TTP Id:

This read-only attribute provides a unique name for an instance of the access Network Management Channel TTP Bidirectional object class within an Access Network (represented by an instance of the managed Element R1 object class)

◊ supported By Object List:

This read/write attribute identifies the supporting resources by pointing to the corresponding equipment, circuit Pack, or software Unit objects.

◊ operational State

This read-only attribute indicates the operability of the S5 flow, i.e., whether it is capable of performing its normal functions.

◊ administrative State

This read/write attribute is used to inhibit (lock) and allow (unlock) the flow of information through the channel.

◊ current Problem List

This read-only attribute is used to indicates the current existing problems, with severity, associated with the channel.

◊ alarm Severity Assignment Profile Pointer

This read/write attribute provides a pointer relationship to an alarm Severity Assignment Profile object.

- Actions:

None

- Notifications:

◊ communications Alarm

This message is used to notify the managing system when a failure has been detected or cleared. The message shall include at least the following information:

* The nature of the alarm (i.e., communications alarm)
* The cause of the failure (e.g., loss of signal)
* The ID of the object reporting the alarm
* The severity of the failure (critical, major, minor, warning, indeterminate, or clear)
* The time and date failure was detected

◊ attribute Value Change

This notification is used to report value changes to the attributes of the object instance.

◊ object Creation

This notification is used to report the creation of an instance of this object class.

◊ object Deletion

This notification is used to report the deletion of an instance of this object class.

◊ state Change

This notification is used to report value changes to the operational State and administrative State attributes.

- Relationship:

  ◊ One instance of this object class exists in the Access Network and is associated with the access Network Management Channel CTP Bidirectional object. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

  ◊ Each instance of this object class is associated with one or more supporting objects (e.g., equipment Holder, circuit Pack, and software Unit). The association is indicated by the supported By Object List attribute of this object and the affected By Object List attribute of the supporting objects.

  ◊ Each instance of this object class is associated with an instance of the alarm Severity Assignment Profile object class. The association is indicated by the alarm Severity Assignment Profile Pointer attribute of the this object.

7.6.4.8 CPE Management Channel CTP Bidirectional

An instance of this object class represents a termination point of a S5 connection that is used for managing a Set Top Box.

- Attributes:

  ◊ cpe Management Channel CTP Id:

  This read-only attribute provides a unique name for an instance of the cpe Management Channel CTP Bidirectional object class within its containing object.

  ◊ frequency:

  For HFC architecture, this read/write attribute indicates the frequency (in KHz) used in carrier modulation. For analog signals, this value indicates the carrier frequency. For digital signals that have been modulated onto an Radio Frequency (RF) carrier, this attribute indicates the center frequency.

  ◊ frequency Spacing:

  For HFC architecture, this read/write attribute indicates the size of the frequency slot in KHz for an RF signal. Typical spacings for video services are 6MHz and 8 MHz.

- Actions:

None

- Notifications:

  ◊ attribute Value Change

  This notification is used to report value changes to the attributes of the object instance.

  ◊ object Creation

  This notification is used to report the creation of an instance of this object class.

  ◊ object Deletion

  This notification is used to report the deletion of an instance of this object class.

- Relationship:

- ◊ At the consumer side of the Access Network, zero or more instances of this object class may be contained in a vc TTP Bidirectional object.
- ◊ At the network side of the Access Network, zero or more instances of this object class may be contained in an vc TTP Bidirectional object.
- ◊ An instance of this object class at the consumer side and an instance of this object class at the network side are associated with each other. The association is indicated by the downstream Connectivity Pointer and upstream Connectivity Pointer attributes of both objects.

7.7 Management Information Details - GDMO & ASN.1 Specification

This sections provides the formal specification of the access network management information using the Guidelines of the Definition of Managed Object (GDMO) and Abstract Syntax Notation One (ASN.1).

7.7.1 Object Classes

7.7.1.1 accessNetworkManagementChannelCTPBidirectional

```
accessNetworkManagementChannelCTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      connectionTerminationPointBidirectional;
  CHARACTERIZED BY accessNetworkManagementChannelCTPBidirectionalPkg PACKAGE
    BEHAVIOUR accessNetworkManagementChannelCTPBidirectionalBeh;
    ATTRIBUTES
      accessNetworkManagementChannelCTPId
        GET;;;
REGISTERED AS {danObjectClass accessNetworkManagementChannelCTPBidirectional(1)};

accessNetworkManagementChannelCTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "An instance of this object class represents a termination point of a S5 connection which is used for managing
the Access Network. One instance of this object class is provided for the Access Network.";
```

7.7.1.2 accessNetworkManagementChannelTTPBidirectional

```
accessNetworkManagementChannelTTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      trailTerminationPointBidirectional;
  CHARACTERIZED BY accessNetworkManagementChannelTTPBidirectionalPkg PACKAGE
    BEHAVIOUR accessNetworkManagementChannelTTPBidirectionalBeh;
    ATTRIBUTES
      accessNetworkManagementChannelTTPId
        GET;;;
  CONDITIONAL PACKAGES
    "ITU Recommendation M.3100":currentProblemListPackage
      PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass accessNetworkManagementChannelTTPBidirectional(2)};

accessNetworkManagementChannelTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "An instance of this object class represents a termination point at which the Access Network S5 information is
originated/formulated and terminated/processed. One instance of this object class exists in an Access Network for
system management.
If the inherited administrativeState attribute is instantiated for an instance of this object class, then setting the
value of administrativeState to unlock allows the termination point to be used. Setting the value of
administrativeState to locked stops the functions of the termination point except for testing, and it immediately
terminates any services using the termination point. Setting the value of administrativeState to shuttingDown,
```


prevents new services from using the termination point and automatically locks it after no active services are using it.";

7.7.1.3 cpeManagementChannelCTPBidirectional

```
cpeManagementChannelCTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      connectionTerminationPointBidirectional;
  CHARACTERIZED BY cpeManagementChannelCTPBidirectionalPkg PACKAGE
    BEHAVIOUR cpeManagementChannelCTPBidirectionalBeh;
    ATTRIBUTES
      cpeManagementChannelCTPId
          GET;;;
  CONDITIONAL PACKAGES
    radioFrequencyPackage
        PRESENT IF "an instance supports it, e.g., in HFC architecture";
REGISTERED AS {danObjectClass cpeManagementChannelCTPBidirectional(3)};

cpeManagementChannelCTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "An instance of this object class represents a termination point of a S5 connection which is used for managing
a Set Top Box. One instance of this object class is provided for each Set Top Box.";
```

7.7.1.4 interfaceInitializationCTPBidirectional

```
interfaceInitializationCTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      connectionTerminationPointBidirectional;
  CHARACTERIZED BY interfaceInitializationCTPBidirectionalPkg PACKAGE
    BEHAVIOUR interfaceInitializationCTPBidirectionalBeh;
    ATTRIBUTES
      interfaceInitializationCTPId
          GET;;;
  CONDITIONAL PACKAGES
    radioFrequencyPackage
        PRESENT IF "an instance supports it, e.g., in HFC architecture";
REGISTERED AS {danObjectClass interfaceInitializationCTPBidirectional(4)};

interfaceInitializationCTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "An instance of this object class represents a bidirectional termination point of the DAVIC Interface
Initialization channel of an access line. The Interface Initialization channel is operated over a specific predefined
VPI/VCI value and is shared by all the Service Consumers of the access line for requesting S3 channel
assignments. The request messages use the DAVIC Interface Initialization Protocol (DIIP). One instance of this
object class is created for each access line at the A1 interface of the Access Network.";
```

7.7.1.5 interfaceInitializationTTPBidirectional

```
interfaceInitializationTTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
    trailTerminationPointBidirectional;
  CHARACTERIZED BY interfaceInitializationTTPBidirectionalPkg PACKAGE
    BEHAVIOUR interfaceInitializationTTPBidirectionalBeh;
    ATTRIBUTES
      interfaceInitializationTTPId
          GET;;;
  CONDITIONAL PACKAGES
    "ITU Recommendation M.3100":currentProblemListPackage
        PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass interfaceInitializationTTPBidirectional(5)};
```

interfaceInitializationTTPBidirectionalBeh BEHAVIOUR
DEFINED AS
"An instance of this object class represents a termination point at which the DAVIC Interface Initialization information is originated/formulated and terminated/processed. One instance of this object class is provided for each access line at the consumer side of the Access Network.
If the inherited administrativeState attribute is instantiated for an instance of this object class, then setting the value of administrativeState to unlock allows the termination point to be used. Setting the value of administrativeState to locked stops the functions of the termination point except for testing, and it immediately terminates any services using the termination point. Setting the value of administrativeState to shuttingDown, prevents new services from using the termination point and automatically locks it after no active services are using it.";

7.7.1.6 multimediaTTPBidirectional

multimediaTTPBidirectional MANAGED OBJECT CLASS
DERIVED FROM "ITU Recommendation M.3100":
trailTerminationPointBidirectional,
multimediaTTPSink,
multimediaTTPSource;
CHARACTERIZED BY multimediaTTPBidirectionalPkg PACKAGE
BEHAVIOUR multimediaTTPBidirectionalBeh;;;
CONDITIONAL PACKAGES
"GR-836 ":loopbackPkg
PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass multimediaTTPBidirectional(6)};

multimediaTTPBidirectionalBeh BEHAVIOUR
DEFINED AS
"For system that supports loopback functions, the operateLoopback and releaseLoopback M-ACTION in the loopbackPkg conditional package provides the capability of instructing the access network to perform and release loopback test on the access line.";

7.7.1.7 multimediaTTPSink

multimediaTTPSink MANAGED OBJECT CLASS
DERIVED FROM "ITU Recommendation M.3100":
trailTerminationPointSink;
CHARACTERIZED BY multimediaTTPSinkPkg PACKAGE
BEHAVIOUR multimediaTTPSinkBeh;
ATTRIBUTES
multimediaTTPId
GET,
accessTopology
GET,
accessMedia
GET,
modulationType
GET-REPLACE;;;
CONDITIONAL PACKAGES
"ITU Recommendation M.3100":currentProblemListPackage
PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass multimediaTTPSink(7)};

multimediaTTPSinkBeh BEHAVIOUR
DEFINED AS
"An instance of this object class is a sink termination point of a multimedia trail. It represents a sink termination point of a physical transmission line across the A1 interface for providing multimedia services.

Information carried on this physical transmission line includes one or more of the following: the S1, S2, S3, S4, and CPE S5 flows and the interface initialization messages (e.g., using DIIP or MAC layer 2 protocol).
If the inherited administrativeState attribute is instantiated for an instance of this object class, then setting the value of administrativeState to unlock allows the termination point to be used. Setting the value of administrativeState to locked stops the functions of the termination point except for testing, and it immediately terminates any services using the termination point. Setting the value of administrativeState to shuttingDown, prevents new services from using the termination point and automatically locks it after no active services are using it.";

7.7.1.8 multimediaTTPSource

```
multimediaTTPSource MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      trailTerminationPointSource;
  CHARACTERIZED BY multimediaTTPSourcePkg PACKAGE
    BEHAVIOUR multimediaTTPSourceBeh;
    ATTRIBUTES
      multimediaTTPId
          GET,
      accessTopology
          GET,
      accessMedia
          GET,
      modulationType
          GET-REPLACE;;;
  CONDITIONAL PACKAGES
    "ITU Recommendation M.3100":currentProblemListPackage
       PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass multimediaTTPSource(8)};

multimediaTTPSourceBeh BEHAVIOUR
 DEFINED AS
   "An instance of this object class is a source termination point of a multimedia trail. It represents a source
termination point of a physical transmission line across the A1 interface for providing multimedia services.
Information carried on this physical transmission line includes one or more of the following: the S1, S2, S3, S4,
and CPE S5 flows and the interface initialization messages (e.g., using DIIP or MAC layer 2 protocol).
If the inherited administrativeState attribute is instantiated for an instance of this object class, then setting the
value of administrativeState to unlock allows the termination point to be used. Setting the value of
administrativeState to locked stops the functions of the termination point except for testing, and it immediately
terminates any services using the termination point. Setting the value of administrativeState to shuttingDown,
prevents new services from using the termination point and automatically locks it after no active services are
using it.";
```

7.7.1.9 serviceRelatedUNChannelCTPBidirectional

```
serviceRelatedUNChannelCTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      connectionTerminationPointBidirectional;
  CHARACTERIZED BY serviceRelatedUNChannelCTPBidirectionalPkg PACKAGE
    BEHAVIOUR serviceRelatedUNChannelCTPBidirectionalBeh;
    ATTRIBUTES
      serviceRelatedUNChannelCTPId
          GET;;;
  CONDITIONAL PACKAGES
    radioFrequencyPackage
       PRESENT IF "an instance supports it, e.g., in HFC architecture";
REGISTERED AS {danObjectClass serviceRelatedUNChannelCTPBidirectional(9)};

serviceRelatedUNChannelCTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
```

"An instance of this object class represents a termination point of a S3 connection. The S3 connection is used for session control and is established using the DIIP messages.";

7.7.1.10 serviceRelatedUNChannelTTPBidirectional

```
serviceRelatedUNChannelTTPBidirectional MANAGED OBJECT CLASS
  DERIVED FROM "ITU Recommendation M.3100":
      trailTerminationPointBidirectional;
  CHARACTERIZED BY serviceRelatedUNChannelTTPBidirectionalPkg PACKAGE
    BEHAVIOUR serviceRelatedUNChannelTTPBidirectionalBeh;
    ATTRIBUTES
      serviceRelatedUNChannelTTPId
        GET;;;
  CONDITIONAL PACKAGES
    "ITU Recommendation M.3100":currentProblemListPackage
      PRESENT IF "an instance supports it";
REGISTERED AS {danObjectClass serviceRelatedUNChannelTTPBidirectional(10)};

serviceRelatedUNChannelTTPBidirectionalBeh BEHAVIOUR
  DEFINED AS
```

"An instance of this object class represents a termination point at which the S3 information is originated/formulated and terminated/processed. Instances of this object class exist in an Access Network if the network side Session control function (Ses) is located in the Access Network. One instance of this object class is provided for each session request using DIIP from a consumer.

If the inherited administrativeState attribute is instantiated for an instance of this object class, then setting the value of administrativeState to unlock allows the termination point to be used. Setting the value of administrativeState to locked stops the functions of the termination point except for testing, and it immediately terminates any services using the termination point. Setting the value of administrativeState to shuttingDown, prevents new services from using the termination point and automatically locks it after no active services are using it.";

7.7.2 Packages

7.7.2.1 radioFrequencyPackage

```
radioFrequencyPackage PACKAGE
  ATTRIBUTES
    frequency
      GET-REPLACE,
    frequencySpacing
      GET-REPLACE;
REGISTERED AS {danPackage radioFrequencyPackage(1)};
```

7.7.3 Attributes

7.7.3.1 accessMedia

```
accessMedia ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.AccessMedia;
  MATCHES FOR EQUALITY;
  BEHAVIOUR accessMediaBeh;
REGISTERED AS {danAttribute accessMedia(1)};

accessMediaBeh BEHAVIOUR
  DEFINED AS
```

"This attribute indicates the access media of an video access line, e.g., twisted pair copper, coax cable, or optical fiber.";

7.7.3.2 accessNetworkManagementChannelCTPId

accessNetworkManagementChannelCTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR accessNetworkManagementChannelCTPIdBeh;
REGISTERED AS {danAttribute accessNetworkManagementChannelCTPId(2)};

accessNetworkManagementChannelCTPIdBeh BEHAVIOUR
 DEFINED AS
  "The accessNetworkManagementChannelCTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the s5anCTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

7.7.3.3 accessNetworkManagementChannelTTPId

accessNetworkManagementChannelTTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR accessNetworkManagementChannelTTPIdBeh;
REGISTERED AS {danAttribute accessNetworkManagementChannelTTPId(3)};

accessNetworkManagementChannelTTPIdBeh BEHAVIOUR
 DEFINED AS
  "The accessNetworkManagementChannelTTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the accessNetworkManagementChannelTTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

7.7.3.4 accessTopology

accessTopology ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.AccessTopology;
  MATCHES FOR EQUALITY;
  BEHAVIOUR accessTopologyBeh;
REGISTERED AS {danAttribute accessTopology(4)};

accessTopologyBeh BEHAVIOUR
 DEFINED AS
  "This attribute indicates the access topology of an video access line, e.g., ADSL, VDSL, FTTC (type 1, 2, etc.), HFC (number of MHz), and FTTH.";

7.7.3.5 associatedCPE

associatedCPE ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0. AssociatedCPE;
  MATCHES FOR EQUALITY;
  BEHAVIOUR associatedCPEBeh;
REGISTERED AS {danAttribute associatedCPE(5)};

associatedCPEBeh BEHAVIOUR
 DEFINED AS
  "This attribute identifies the CPE that is associated the service session.";

7.7.3.6 cpeManagementChannelCTPId

cpeManagementChannelCTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR cpeManagementChannelCTPIdBeh;

REGISTERED AS {danAttribute cpeManagementChannelCTPId(6)};

cpeManagementChannelCTPIdBeh BEHAVIOUR
 DEFINED AS
  "The cpeManagementChannelCTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the s5stbCTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

7.7.3.7 frequency

frequency ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.Frequency;
  MATCHES FOR EQUALITY, ORDERING;
  BEHAVIOUR frequencyBeh;
REGISTERED AS {danAttribute frequency(7)};

frequencyBeh BEHAVIOUR
 DEFINED AS
  "This attribute indicates the frequency in KHz used in carrier modulation. For analog video signals, this value identifies the video carrier frequency. For digital signals which have been modulated onto an RF carrier, this attribute identifies the center frequency.";

7.7.3.8 frequencySpacing

frequencySpacing ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.Integer;
  MATCHES FOR EQUALITY, ORDERING;
  BEHAVIOUR frequencySpacingBeh;
REGISTERED AS {danAttribute frequencySpacing(8)};

frequencySpacingBeh BEHAVIOUR
 DEFINED AS
  "This attribute indicates the size of the frequency slot in KHz for an radio frequency (RF) signal. Typical spacings for video services are 6 MHz and 8 MHz.";

7.7.3.9 interfaceInitializationCTPId

interfaceInitializationCTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR interfaceInitializationCTPIdBeh;
REGISTERED AS {danAttribute interfaceInitializationCTPId(9)};

interfaceInitializationCTPIdBeh BEHAVIOUR
 DEFINED AS
  "The interfaceInitializationCTPId attribute is an attribute type whose distinguished value can be used as a RDN when an instance of the interfaceInitializationCTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

7.7.3.10 interfaceInitializationTTPId

interfaceInitializationTTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR interfaceInitializationTTPIdBeh;
REGISTERED AS {danAttribute interfaceInitializationTTPId(10)};

interfaceInitializationTTPIdBeh BEHAVIOUR

DEFINED AS
"The interfaceInitializationTTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the interfaceInitializationTTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

7.7.3.11 modulationType

```
modulationType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.ModulationType;
  MATCHES FOR EQUALITY;
  BEHAVIOUR modulationTypeBeh;
REGISTERED AS {danAttribute modulationType(11)};

modulationTypeBeh BEHAVIOUR
 DEFINED AS
  "This attribute specifies the digital modulation scheme used to modulate a digital stream for analog transmission.";
```

7.7.3.12 multimediaTTPId

```
multimediaTTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR multimediaTTPIdBeh;
REGISTERED AS {danAttribute multimediaTTPId(12)};

multimediaTTPIdBeh BEHAVIOUR
 DEFINED AS
  "The multimediaTTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the multimediaTTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";
```

7.7.3.13 s3ChannelType

```
s3ChannelType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.S3ChannelType;
  MATCHES FOR EQUALITY;
  BEHAVIOUR s3ChannelTypeBeh;
REGISTERED AS {danAttribute s3ChannelType(13)};

s3ChannelTypeBeh BEHAVIOUR
 DEFINED AS
  "The s3ChannelType attribute indicates the type of the S3 flow. Possible channel types include DSMCC Channel Change channel, UN Signaling channel, and UN Configuration channel.";
```

7.7.3.14 serviceRelatedUNChannelCTPId

```
serviceRelatedUNChannelCTPId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
  MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
  BEHAVIOUR serviceRelatedUNChannelCTPIdBeh;
REGISTERED AS {danAttribute serviceRelatedUNChannelCTPId(14)};

serviceRelatedUNChannelCTPIdBeh BEHAVIOUR
 DEFINED AS
  "The serviceRelatedUNChannelCTPId attribute is an attribute type whose distinguished value can be used as a RDN when naming an instance of the s3CTPBidirectional object class. If the pString choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";
```

7.7.3.15 serviceRelatedUNChannelTTPId

```
 serviceRelatedUNChannelTTPId ATTRIBUTE
   WITH ATTRIBUTE SYNTAX DavicAccessNetworkModV0.NameType;
   MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
   BEHAVIOUR serviceRelatedUNChannelTTPIdBeh;
REGISTERED AS {danAttribute serviceRelatedUNChannelTTPId(15)};

serviceRelatedUNChannelTTPIdBeh BEHAVIOUR
  DEFINED AS
    "The serviceRelatedUNChannelTTPId attribute is an attribute type whose distinguished value can be used as a
RDN when naming an instance of the serviceRelatedUNChannelTTPBidirectional object class. If the pString
choice for the syntax is used, then matching on the substrings is permitted. If the numericName choice for the
syntax is used, then matching on ordering is permitted.";
```

7.7.4 Name Bindings

7.7.4.1 accessNetworkManagementChannelCTPBidirectional-vcTTPBidirectional

```
accessNetworkManagementChannelCTPBidirectional-vcTTPBidirectional NAME BINDING
   SUBORDINATE OBJECT CLASS accessNetworkManagementChannelCTPBidirectional AND
SUBCLASSES;
   NAMED BY SUPERIOR OBJECT CLASS "ATM Forum M4 NE View":
   vcTTPBidirectional AND SUBCLASSES;
   WITH ATTRIBUTE accessNetworkManagementChannelCTPId;
   BEHAVIOUR accessNetworkManagementChannelCTPBidirectional-vcTTPBidirectionalBeh;
REGISTERED AS {danNameBinding
   accessNetworkManagementChannelCTPBidirectional-vcTTPBidirectional(1)};

accessNetworkManagementChannelCTPBidirectional-vcTTPBidirectionalBeh BEHAVIOUR
  DEFINED AS
    "The subordinate object shall be automatically instantiated or deleted.";
```

7.7.4.2 accessNetworkManagementChannelTTPBidirectional-managedElement

```
accessNetworkManagementChannelTTPBidirectional-managedElement NAME BINDING
   SUBORDINATE OBJECT CLASS accessNetworkManagementChannelTTPBidirectional AND
SUBCLASSES;
   NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":
   managedElement AND SUBCLASSES;
   WITH ATTRIBUTE accessNetworkManagementChannelTTPId;
   BEHAVIOUR accessNetworkManagementChannelTTPBidirectional-managedElementBeh;
REGISTERED AS {danNameBinding accessNetworkManagementChannelTTPBidirectional-
managedElement(2)};

accessNetworkManagementChannelTTPBidirectional-managedElementBeh BEHAVIOUR
  DEFINED AS
    "The subordinate object shall be automatically instantiated or deleted.";
```

7.7.4.3 cpeManagementChannelCTPBidirectional-interworkingVCTTPBidirectional

```
cpeManagementChannelCTPBidirectional-interworkingVCTTPBidirectional NAME BINDING
   SUBORDINATE OBJECT CLASS cpeManagementChannelCTPBidirectional AND SUBCLASSES;
   NAMED BY SUPERIOR OBJECT CLASS "ATM Forum 96-0383R3":
   interworkingVCTTPBidirectional AND SUBCLASSES;
   WITH ATTRIBUTE cpeManagementChannelCTPId;
   BEHAVIOUR cpeManagementChannelCTPBidirectional-interworkingVCTTPBidirectionalBeh;
   CREATE
     WITH-REFERENCE-OBJECT,
     WITH-AUTOMATIC-INSTANCE-NAMING;
```



```
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding
  cpeManagementChannelCTPBidirectional-interworkingVCTTPBidirectional(3)};

cpeManagementChannelCTPBidirectional-interworkingVCTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
  "The subordinate object may be: 1) automatically instantiated/deleted according to the mode of operation of the
Access Network, or (2) explicitly instantiated/deleted by system management operation.";
```

7.7.4.4 cpeManagementChannelCTPBidirectional-multimediaTTPBidirectional

```
cpeManagementChannelCTPBidirectional-multimediaTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS cpeManagementChannelCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS multimediaTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE cpeManagementChannelCTPId;
  BEHAVIOUR cpeManagementChannelCTPBidirectional-multimediaTTPBidirectionalBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding
  cpeManagementChannelCTPBidirectional-multimediaTTPBidirectional(4)};

cpeManagementChannelCTPBidirectional-multimediaTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
  "The subordinate object may be: 1) automatically instantiated/deleted according to the mode of operation of the
Access Network, or (2) explicitly instantiated/deleted by system management operation.";
```

7.7.4.5 cpeManagementChannelCTPBidirectional-vcTTPBidirectional

```
cpeManagementChannelCTPBidirectional-vcTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS cpeManagementChannelCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ATM Forum M4 NE View":
  vcTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE cpeManagementChannelCTPId;
  BEHAVIOUR cpeManagementChannelCTPBidirectional-vcTTPBidirectionalBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding
  cpeManagementChannelCTPBidirectional-vcTTPBidirectional(5)};

cpeManagementChannelCTPBidirectional-vcTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
  "The subordinate object may be: 1) automatically instantiated/deleted according to the mode of operation of the
Access Network, or (2) explicitly instantiated/deleted by system management operation.";
```

7.7.4.6 interfaceInitializationCTPBidirectional-multimediaTTPBidirectional

```
interfaceInitializationCTPBidirectional-multimediaTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS interfaceInitializationCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS multimediaTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE interfaceInitializationCTPId;
  BEHAVIOUR interfaceInitializationCTPBidirectional-multimediaTTPBidirectionalBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
```

```
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding
  interfaceInitializationCTPBidirectional-multimediaTTPBidirectional(6)};

interfaceInitializationCTPBidirectional-multimediaTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
  "The subordinate object may be: 1) automatically instantiated/deleted according to the mode of operation of the
Access Network, or (2) explicitly instantiated/deleted by system management operation. The subordinate
managed object is automatically deleted when the superior managed object is deleted.";
```

7.7.4.7 interfaceInitializationCTPBidirectional-vcTTPBidirectional

```
interfaceInitializationCTPBidirectional-vcTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS interfaceInitializationCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ATM Forum M4 NE View":
  vcTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE interfaceInitializationCTPId;
  BEHAVIOUR interfaceInitializationCTPBidirectional-vcTTPBidirectionalBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding
  interfaceInitializationCTPBidirectional-vcTTPBidirectional(7)};

interfaceInitializationCTPBidirectional-vcTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
  "For point to point configuration, the subordinate managed object may be: (1) automatically instantiated when
the predefined VP and VC for interface initialization is established or (2) explicitly created by system
management operation. The subordinate managed object is automatically deleted when the superior managed
object is deleted.";
```

7.7.4.8 interfaceInitializationTTPBidirectional-managedElement

```
interfaceInitializationTTPBidirectional-managedElement NAME BINDING
  SUBORDINATE OBJECT CLASS interfaceInitializationTTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":
  managedElement AND SUBCLASSES;
  WITH ATTRIBUTE interfaceInitializationTTPId;
  BEHAVIOUR interfaceInitializationTTPBidirectional-managedElementBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding interfaceInitializationTTPBidirectional-managedElement(8)};

interfaceInitializationTTPBidirectional-managedElementBeh BEHAVIOUR
 DEFINED AS
  "For point to point configuration, the subordinate managed object may be: (1) automatically instantiated when
the predefined VP and VC for interface initialization is established or (2) explicitly created by system
management operation. The subordinate managed object is automatically deleted when the superior managed
object is deleted.";
```

7.7.4.9 multimediaTTPBidirectional-managedElement

```
multimediaTTPBidirectional-managedElement NAME BINDING
```



```
  SUBORDINATE OBJECT CLASS multimediaTTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":
  managedElement AND SUBCLASSES;
  WITH ATTRIBUTE multimediaTTPId;
  BEHAVIOUR multimediaTTPBidirectional-managedElementBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS {danNameBinding multimediaTTPBidirectional-managedElement(9) };

multimediaTTPBidirectional-managedElementBeh BEHAVIOUR
 DEFINED AS
   "The subordinate object may be: (1) automatically instantiated/deleted according to the mode of operation of
the Access Network, or (2) explicitly instantiated/deleted by system management operation.";
```

7.7.4.10 serviceRelatedUNChannelCTPBidirectional-interworkingVCTTPBidirectional

```
serviceRelatedUNChannelCTPBidirectional-interworkingVCTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS serviceRelatedUNChannelCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ATM Forum 96-0383R3":
  interworkingVCTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE serviceRelatedUNChannelCTPId;
  BEHAVIOUR serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectionalBeh;
REGISTERED AS {danNameBinding serviceRelatedUNChannelCTPBidirectional-
interworkingVCTTPBidirectional(10)};

serviceRelatedUNChannelCTPBidirectional-interworkingVCTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "The subordinate managed object is automatically instantiated according to the S3 assignment request (through
MAC layer 2 protocol) from the Set Top Unit. The subordinate managed object is automatically deleted
according to the S3 signaling.";
```

7.7.4.11 serviceRelatedUNChannelCTPBidirectional-multimediaTTPBidirectional

```
serviceRelatedUNChannelCTPBidirectional-multimediaTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS serviceRelatedUNChannelCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS multimediaTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE serviceRelatedUNChannelCTPId;
  BEHAVIOUR serviceRelatedUNChannelCTPBidirectional-multimediaTTPBidirectionalBeh;
REGISTERED AS {danNameBinding serviceRelatedUNChannelCTPBidirectional-
multimediaTTPBidirectional(11)};

serviceRelatedUNChannelCTPBidirectional-multimediaTTPBidirectionalBeh BEHAVIOUR
 DEFINED AS
   "The subordinate managed object is automatically instantiated according to the S3 assignment request (through
MAC layer 2 protocol) from the Set Top Unit. The subordinate managed object is automatically deleted
according to the S3 signaling.";
```

7.7.4.12 serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectional

```
serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectional NAME BINDING
  SUBORDINATE OBJECT CLASS serviceRelatedUNChannelCTPBidirectional AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ATM Forum M4 NE View":
  vcTTPBidirectional AND SUBCLASSES;
  WITH ATTRIBUTE serviceRelatedUNChannelCTPId;
  BEHAVIOUR serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectionalBeh;
REGISTERED AS {danNameBinding serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectional(12)};
```

serviceRelatedUNChannelCTPBidirectional-vcTTPBidirectionalBeh BEHAVIOUR
DEFINED AS
"The subordinate managed object is automatically instantiated according to the S3 assignment request (through DIIP) from the Set Top Unit. The subordinate managed object is automatically deleted according to the S3 signaling.";

7.7.4.13 serviceRelatedUNChannelTTPBidirectional-managedElement

serviceRelatedUNChannelTTPBidirectional-managedElement NAME BINDING
SUBORDINATE OBJECT CLASS serviceRelatedUNChannelTTPBidirectional AND SUBCLASSES;
NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":
managedElement AND SUBCLASSES;
WITH ATTRIBUTE serviceRelatedUNChannelTTPId;
BEHAVIOUR serviceRelatedUNChannelTTPBidirectional-managedElementBeh;
REGISTERED AS {danNameBinding serviceRelatedUNChannelTTPBidirectional-managedElement(13)};

serviceRelatedUNChannelTTPBidirectional-managedElementBeh BEHAVIOUR
DEFINED AS
"If the Session control function is located in the Access Network, then the subordinate managed object is automatically instantiated according to the S3 assignment request (through DIIP) from the Set Top Unit. The subordinate managed object is automatically deleted according to the S3 signaling.";

7.7.5 ASN.1 Productions

DavicAccessNetworkModV0 {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) accessNetworkInformationModel(5) asn1Module(2) davicAccessNetworkModV0(0)}

```
DEFINITIONS IMPLICIT TAGS EXTENSIBILITY IMPLIED ::=
BEGIN

  -- EXPORTS everything

  IMPORTS

  NameType
  FROM
  ASN1DefinedTypesModule {ccitt recommendation m gnm(3100)
    informationModel(0) asn1Modules(2) asn1DefinedTypesModule(0)};

  -- Object Identifiers

  davic  OBJECT IDENTIFIER ::= {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)}

  davicAccessNetwork OBJECT IDENTIFIER ::= {davic accessNetworkInformationModel(5)}

  danStandardSpecificExtension OBJECT IDENTIFIER ::= {davicAccessNetwork
        standardSpecificExtension(0)}

  danASN1Module OBJECT IDENTIFIER ::= {davicAccessNetwork asn1Module(2)}

  danObjectClass OBJECT IDENTIFIER ::= {davicAccessNetwork managedObjectClass(3)}

  danPackage OBJECT IDENTIFIER ::= { davicAccessNetwork package(4)}

  danParameter OBJECT IDENTIFIER ::= { davicAccessNetwork parameter(5)}

  danNameBinding OBJECT IDENTIFIER ::= { davicAccessNetwork nameBinding(6)}

  danAttribute OBJECT IDENTIFIER ::= { davicAccessNetwork attribute(7)}
```



```
danAttributeGroup OBJECT IDENTIFIER ::= { davicAccessNetwork attributeGroup(8)}

danAction OBJECT IDENTIFIER ::= { davicAccessNetwork action(9)}

danNotification OBJECT IDENTIFIER ::= { davicAccessNetwork notification(10)}


-- supporting productions

AccessMedia ::= ENUMERATED {
  copper        (0),
  coax (1),
  fiber (2)}

AccessTopology ::= CHOICE {
  adsl          [0] NULL, -- ADSL
  vdsl          [1] NULL, -- VDSL
  ftth          [2] NULL, -- FTTH
  fttc          [3] INTEGER, -- Type of FTTC, e.g., types 1, 2, 3, etc.
  hfc           [4] INTEGER -- HFC, Number of MHz, e.g., 550 MHz, 1000 MHz, etc.
  }

AssociatedCPE ::= SEQUENCE {
  consumerName          GraphicString OPTIONAL,
  osiNsapAddress        OCTET STRING (SIZE (20))}   -- 20 bytes, as defined in ISO/IEC 13818-6

Frequency ::= CHOICE {
  centerFrequency               [1] INTEGER,
  videoCarrierFrequency         [2] INTEGER}

Integer ::= INTEGER

ModulationType ::= CHOICE {
  common                ENUMERATED {
        qam16   (0),
        qam64   (1),
        qam256  (2),
        vsb16   (3),
        qpsk    (4),
        bpsk    (5),
        cap4    (6),
        cap16   (7),
        ofdm    (8)},
  specific              PrintableString,
  unknown               [0] NULL,
  notApplicable         [1] NULL}

S3ChannelType ::= ENUMERATED {
  dsmccChannelChangeChannel   (0),
  uNSignalingChannel          (1),
  uNConfigurationChannel      (2)}


END -- end of productions
```

8 Service Related Control Management Information Model

8.1 Introduction

This Clause defines service-related management information that will allow DAVIC Delivery Systems to support broadcast and video-on-demand services. The main components of the Delivery System are shown in Figure 8-1. It specifies a CMIP-based information model with a GDMO/ASN.1 specification. The general model presented here is applicable to any Delivery System based on the Switched Digital Video-based Access Network. It has been assumed an ATM-based Fiber-to-the-Curb (FTTC) or Hybrid Fiber Coax (HFC) Access Network. Other network technologies may require model extensions.

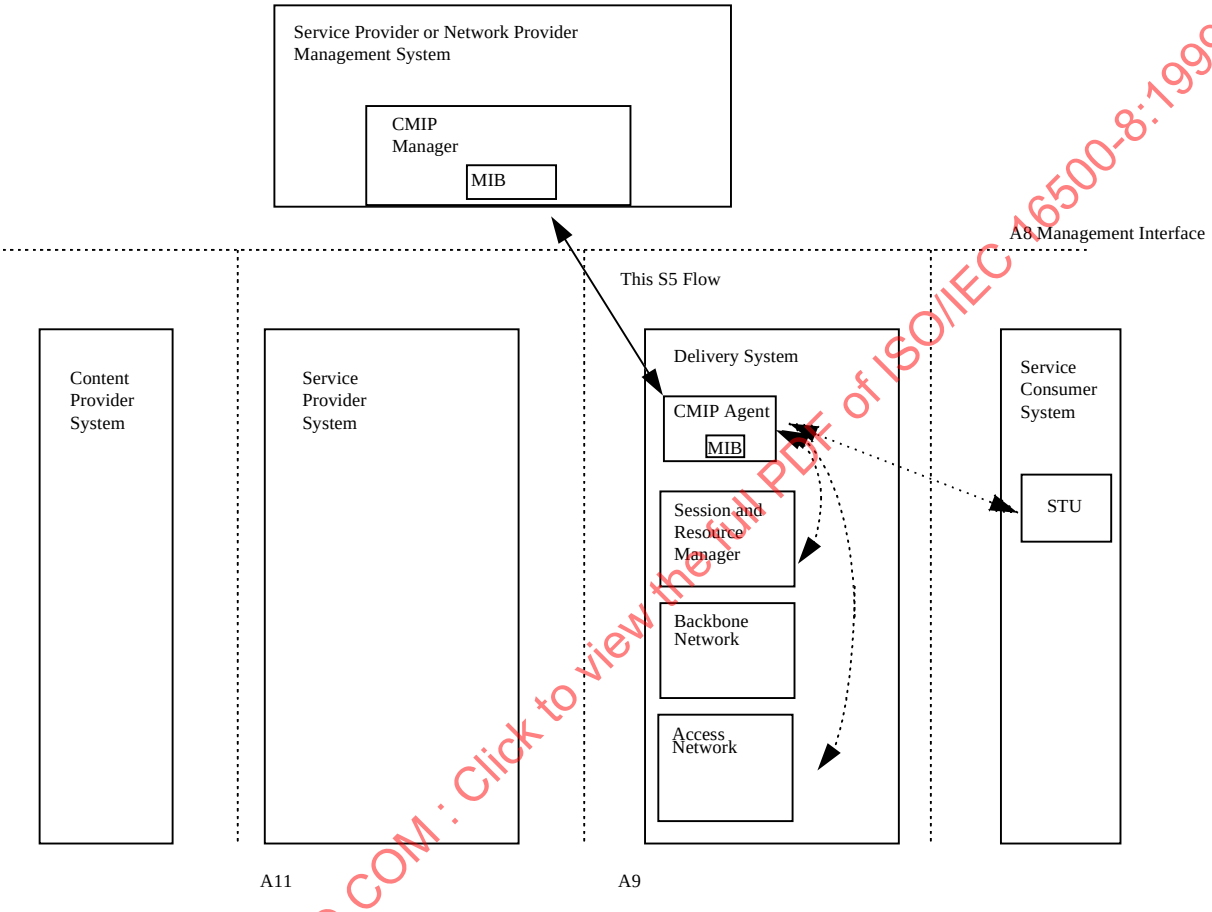


Figure 8-1 — Context of the Management Interface

Motivation for this management interface is presented in subclause 8.2. An overview of the management data for this interface is provided in subclause 8.3. Subclause 8.4 provides an object-oriented Management Information Model (MIM) for this interface using the Guidelines for the Definition of Managed Objects (GDMO).

8.2 Information Model Motivation

The data in this information model represents resources in the Delivery System that are used to manage subscriber service. These resources are used to configure the subscriber's equipment at service initiation, to enable the subscriber to access the services they are authorized for, and to deny them access to unauthorized service. The following figure describes the overall architecture and the interfaces to the Delivery System.

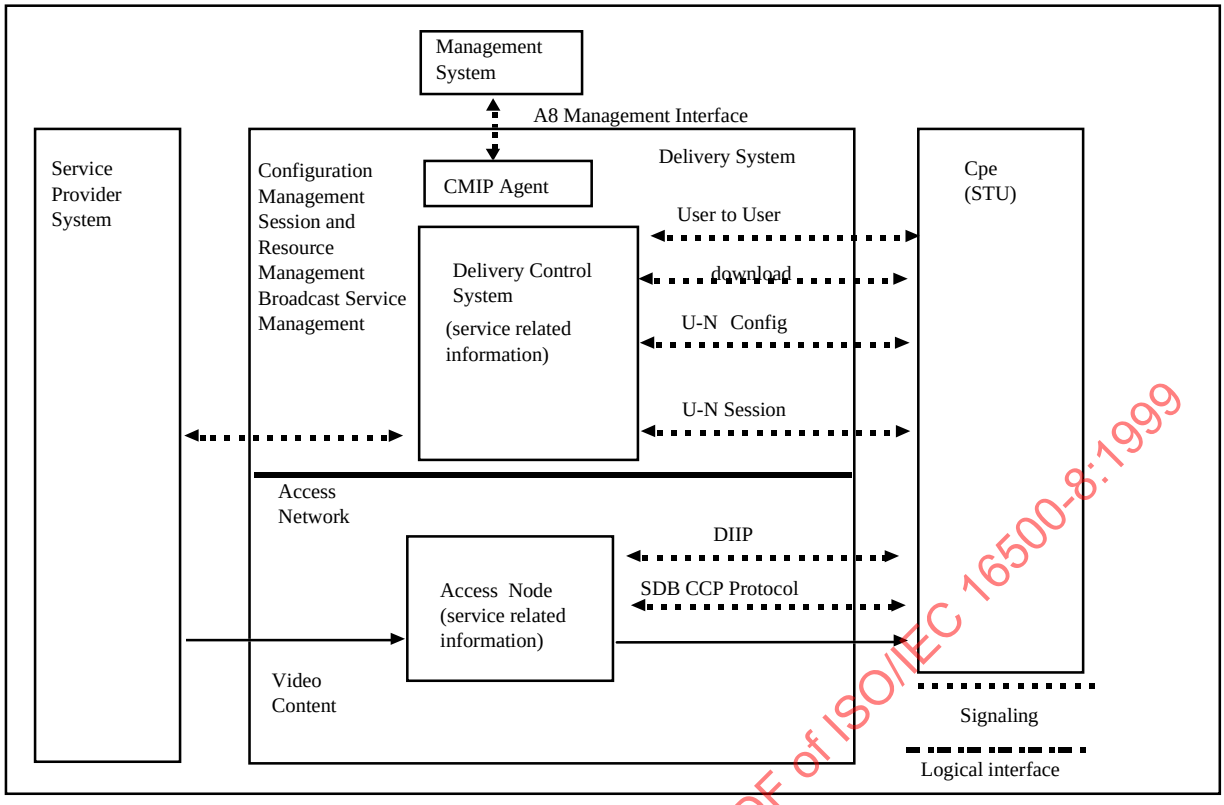


Figure 8-2 — Overview of the Functions and Protocols Supported by the Delivery System

The following functions/protocols can be used to motivate the resources in this information model:

- The DAVIC Interface Initialization Protocol (DIIP), which is used to inform the CPE of network signaling addresses terminated within the Delivery System (typically within the Access Node).
- The session, resource & configuration management function should be supported by the Delivery System. This implies that the Delivery System supports the DSM-CC User-Network protocol which includes UNConfig messages that assign the configuration server for the CPE and UNSessionSetup messages through which a CPE requests connection to a particular service. It also includes the Download protocol that allows the Cpe to download boot information.
- The Delivery System in this model also supports the overall management of the end-to-end broadcast service, which implies that it should also terminate "user-to-user" signaling from the CPE.
- The Delivery System should also terminate the DSM-CC Switched Digital Broadcasting (SDB) Channel Change Protocol (CCP) through which a CPE requests to be switched to a different broadcast channel.

8.2.1 Configuration Management

The following figure shows the entities supporting configuration management in the Delivery Network. Although not strictly required, the DIIP Server will be typically resident within the Access Node within the Delivery System. The UNConfig Server and the Download Server providing the Cpe with its configuration and boot information are typically a part of the Delivery Control System. Note that only the service-related information relevant to these functions will be addressed in the information model.

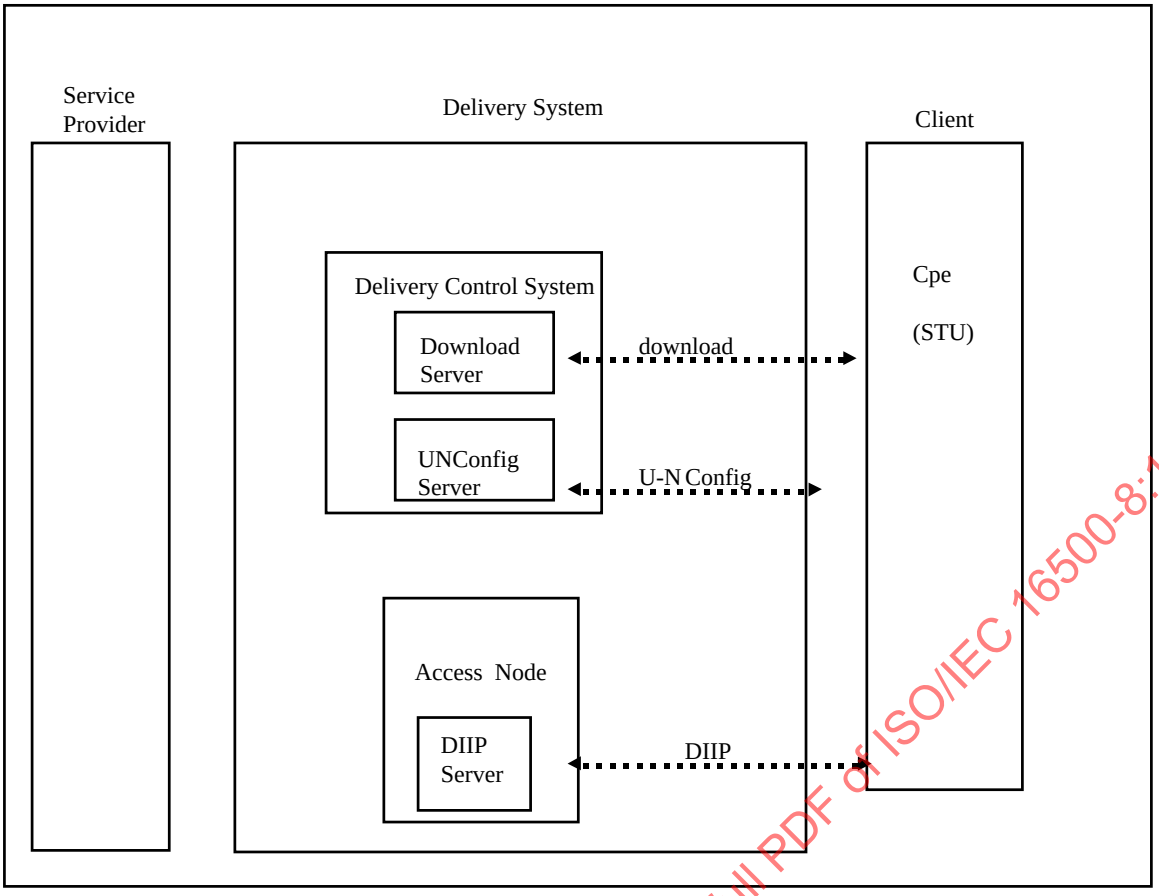


Figure 8-3 — Overview of the System Components Required for Configuration Management

8.2.1.1 DAVIC Interface Initialization Protocol

The DAVIC Interface Initialization Protocol (DIIP) is used to assign the initial parameters at the CPE that allow connections (channels) from the CPE to signaling entities in the Delivery System (e.g. SESn, BCU-BroadcastControlUnit). If the DIIP is supported, the STB initiates the protocol by sending an ASSIGN_REQUEST message to the Delivery System. The CPE identifies its own deviceId in the message and the type of channel connection it needs. The deviceId is the MACId of the CPE. The channelType being requested can be either: a Session Control channel for S3 DSM-CC U-N signaling; a STU configuration channel for DSM-CC U-N Configuration, or a channel for Switched Broadcast Control for DSM-CC SDB-CCP signaling. Note that at the service layer the DIIP Server needs information required to validate a Cpe based on its deviceId. The Connection information required may not be modeled at the service layer.

The Network will validate the deviceId and the requested channel type against the subscriber authorization information and will respond with the requested channels or deny the request. For PVC connections the Network will provide the VPI and VCI for the connection. For SVC connections the Network will provide the NSAP address of the connection entity.

The use of this protocol motivates the following data to be available in the Delivery System prior to CPE initialization:

- data that describes the signaling connections in the Delivery System. This data is represented by the controlSystem managed object and includes signaling address and the service area domain for the signaling connection.
- subscriber authorizations for different signaling connection types. This data is represented by the subscription data in the broadbandDrop managed object.
- a list of authorized CPE devices. This data is represented by the deviceId attribute in the cpe managed object.

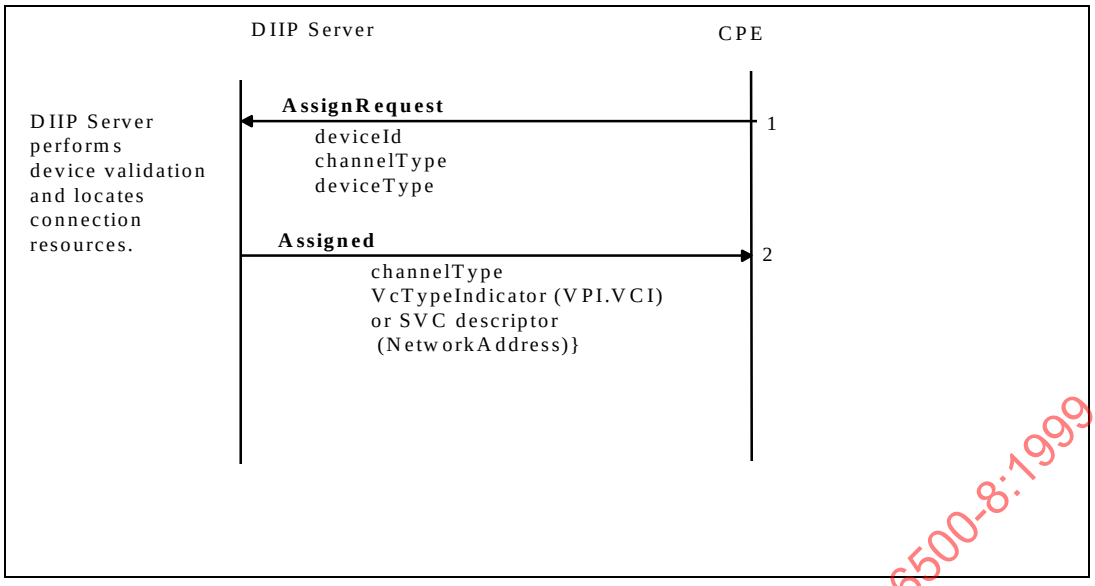


Figure 8-4 — Overview of the DIIP

8.2.1.2 User-Network Configuration

To obtain its configuration parameters, a User device sends a UNConfigRequest message to the Network.. The User device request optionally includes an identification of the device, model and version. Using this information, the Network determines the appropriate configuration parameters for that device and sends those parameters to the User device in the UNConfigConfirm message. This data includes a UserId, which contains the NSAP that is used to uniquely identify the User on the Network and a ServerId, which contains the NSAP that identifies the default Server for this user. This will be the address of the Server, which supplies initial download and applications to the User device.

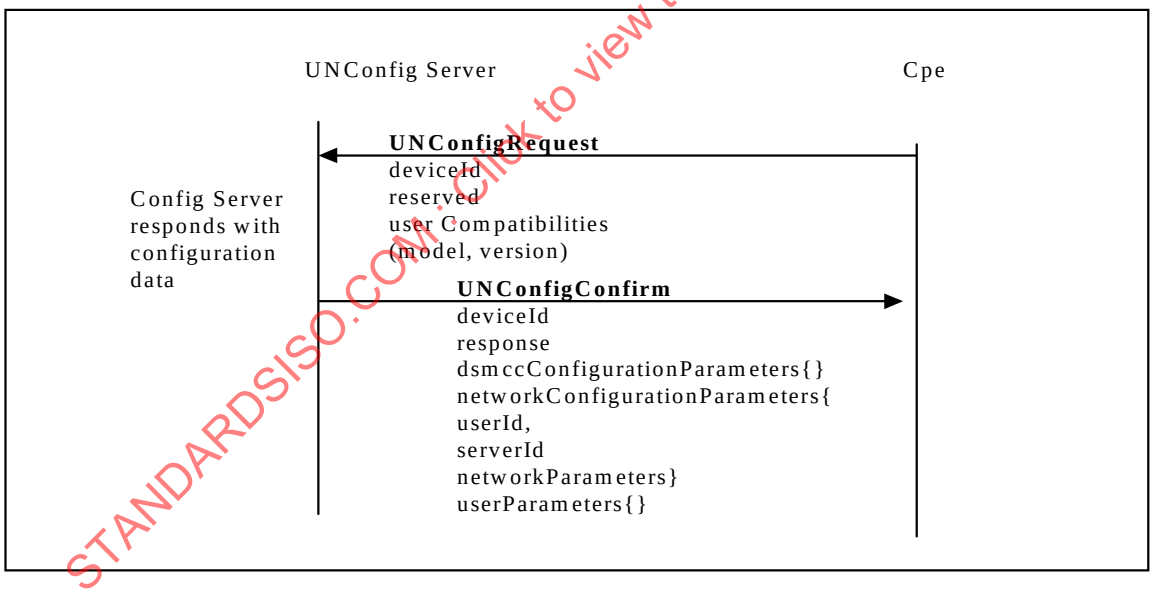


Figure 8-5 — Details from the UNConfig Protocol

8.2.1.2.1 Information Requirements for the User-Network Configuration Protocol

To support the UNConfig protocol the following data is needed in the Service Management Layer (SML) related to the User device:

- User Id (NSAP address for the User). In the information model this is represented by the combination of the broadbandDropId (E.164 address), cpeAddress (ipAddress) and cpeDeviceId (MAC id).
- DeviceId of the User device. This is the cpeDeviceId in the cpe managed object.
- ServerId for the boot Server. This is represented by an instance of the serviceConnectionPoint managed object.
- Model and version of the User device. This is represented by the optional cpeType field in the cpe managed object.

8.2.2 Video on Demand Service-Related Control

The following figure indicates the main system components required for the interactive video services.

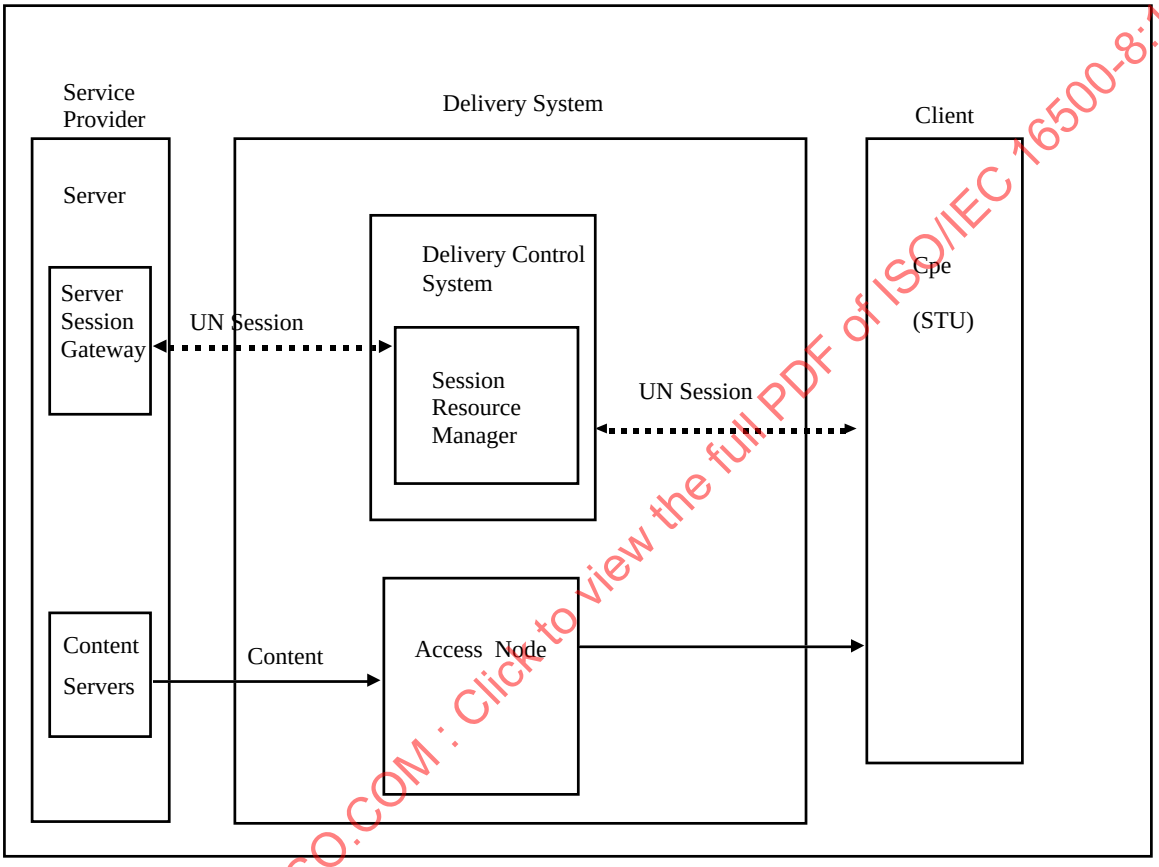


Figure 8-6 — Overview of the Main System Components in a VOD System

For video on demand (or other on-demand service) the Session Manager and CPE use DSM/CC user-to-network signaling to set up the Session connection. The CPE sends a ClientSessionSetUpRequest which includes a SessionId (which is composed of the CPE Device Id and a sessionNumber), a ClientId (which is the NSAP address of the User), and the ServerId of the desired Video Service. (The SessionId may also be assigned by the network.) The Session Manager will validate the ClientId and the requested ServerId and will determine if the User is authorized for the desired Service. Note that this protocol could be used to set up to other types of on-demand services besides video on demand, such as on-demand Internet access, or on-demand pay-per-view authorizations. If all validations pass, the Session Manager will signal the Service Provider System about the User request.

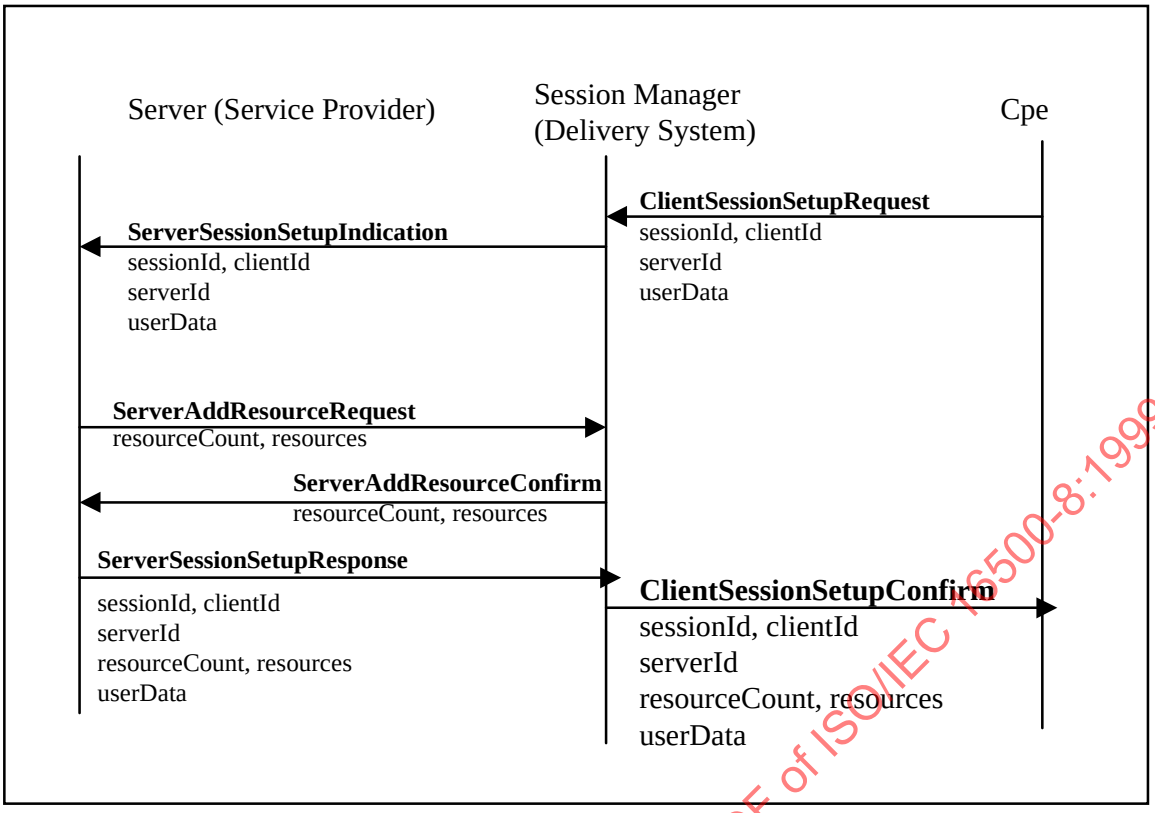


Figure 8-7 — UN Session Setup and Confirm

8.2.2.1 Information Requirements for the User-Network Session Setup Protocol

The use of the U-N Session Protocol motivates a number of data elements to be stored in the Session Manager:

- CPE Device Id. This is represented in the cpe managed object.
- ServerId. This is represented in the serviceConnectionPoint managed object.
- Server IP address. This is represented in the serviceConnectionPoint managed object.
- Service type associated with a ServerId. This is represented in the serviceConnectionPoint managed object.
- CPE NSAP Address. This is represented by the broadbandDropId (E.164) in the broadbandDrop managed object and the cpeDeviceId and cpeAddress in the cpe managed object.
- CPE IP Address. This is represented by the cpeAddress of the cpe managed object.

8.2.3 Broadcast Service Management in the Delivery System Domain

The DSM-CC Annex H provides an overview of how an end-to-end broadcast service may be managed in a network supporting the Switched Digital Video (SDV) architecture. The key distinguishing feature in an SDV architecture is the way in which broadcast services are managed. For broadcast services in a SDV architecture, the bandwidth made available to each end-user is typically limited to a single broadcast program. However, the end-user is allowed access to a larger selection of programs by *switching* the desired broadcast program to the end-user upon request. The broadcast programs are available at a point in the Delivery Network known as the Replication Unit. The following figure details the main system components required for setting up an end-to-end broadcast service, in an SDV architecture. The terminology used, however, is slightly different from that used within Annex H.

From the Cpe viewpoint, the protocols used for accessing a broadcast server are similar to those used for accessing an interactive service. The key difference is that the “broadcast server” is distributed. In addition to the SDB Management Server, which terminates the server session-side signaling as well as the user-to-user protocol from a Cpe, a server function represented by the BCU is also present (typically within the Access Node). Although the BCU should be resident within the delivery network, the SDB Management Server is not required

to be a part of the Delivery Network. The information model presented here, however, assumes that the Delivery Network supports the SDB Management Server. This allows the Delivery Network to present a unified view of the broadcast service offerings available from various service providers.

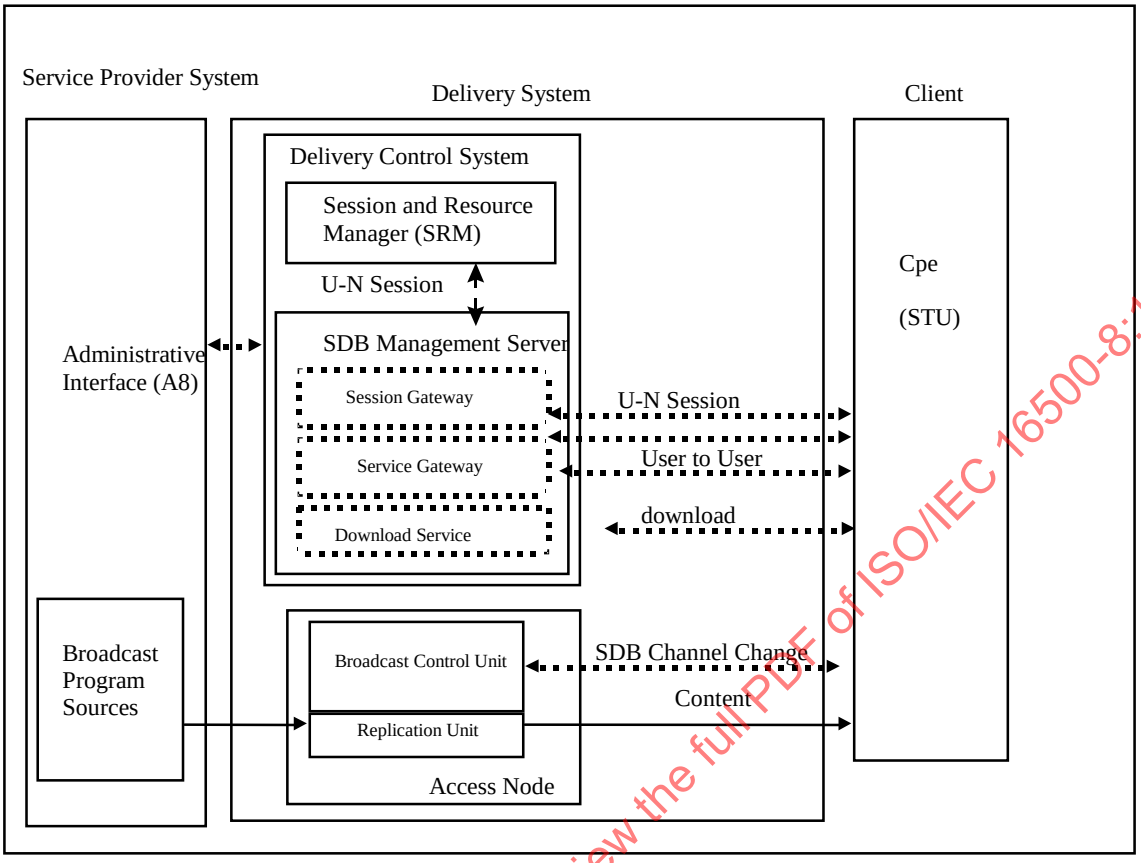


Figure 8-8 — Overview of the Main System Components Required for Broadcast Service

After a Cpe has signed on with the network and received initial configuration information, it should establish a broadcast session to receive broadcast service. The Cpe can establish a broadcast session with a Switched Broadcast (SDB) Management Server using the standard DSM-CC UN Session protocol. In simpler implementations, explicit session establishment is not required and default sessions may be used. The established session provides the Cpe with the resources needed to receive the broadcast service. These resources include amongst others, a signaling channel to the BCU (see DSM-CC Annex H for further details). Although the SDB Management Server could reside in the Service Provider's domain, we assume a case in which the SDB Management Server resides within the Delivery System and provides a unified view of the broadcast services available to the Cpe. It is possible that some of the components required by the SDB Management Server are implemented outside the Delivery System; here we show all logical entities required to support the overall broadcast service.

After session establishment with the SDB Management Server, the Client should download the necessary program guide information, which describe the overall service offering to the Cpe. The program guide information shouldinclude the mapping between the network view of the channel, described by the "broadcastProgramId, and the "user perceived channel identifier", in addition to other information such as "call signs"of the Service Provider. The program guide information is downloaded to the Client using the DSM-CC Download Protocol. The Download Server supports the download protocol. Note that although the Download Service may be part of the SRM within the Delivery System (as depicted in Figure 8-8), it could also be resident outside in the Service Provider's domain. If the Delivery System provides this download service for broadcast channels, additional managed objects would need to be supported beyond those specified in this document.

The Cpe can then request to watch a particular broadcast channel using the SDB Channel Change protocol. The Channel Change Protocol is terminated by the Broadcast Control Unit (BCU). Note again that although the figure shows the BCU to be a part of the Access Node, the BCU function may easily be located elsewhere within the Delivery Network.

A broadcast service offering would typically consist of offering pay-per-view in addition to subscription-based services. The Service Gateway within the SDB Management server would terminate the user-user signaling required for purchasing such events.

If the Delivery System supports the SDB Management Server then the following information shouldbe made available in the Delivery System.

- An interface to the SDB Management Server shouldbe represented.
- A notion of broadcast programs, which associate the physical connections available to carry a broadcast channel with a logical identifier known as the broadcastProgramId.
- A notion of “service events”, such as “pay-per-view” events that are part of the broadcast service.
- A logical notion of “service domains”. A service domain identifies the Cpe population which will receive a particular set of service offerings.
- A notion of client service profile, which associates a particular Cpe with the Cpe entitlements to various broadcast services

Even if the SDB Management Server is not supported within the Delivery System, the BCU shouldbe present within the Delivery System and it shouldsupport service-related information in order to terminate the SDB Channel Change protocol. Typically the BCU will be a part of the Access Node within the Delivery System. As the service information required by the BCU is only a small subset of the information required by the SDB Management Server, it is helpful to identify the information requirement of the BCU separately. The following section motivates the service information requirements at the BCU.

8.2.3.1 Service-Related Information Requirements at the BCU

As mentioned earlier, in a SDV broadcast service architecture, the bandwidth made available to each end-user is typically limited to a single broadcast program. However, the end-user is allowed access to a larger selection of programs by *switching* the desired broadcast program to the end-user upon request. The broadcast programs are available at a point in the Delivery Network known as the Replication Unit. DAVIC specifies the DSM-CC Switched Digital Broadcasting (SDB) Channel Change Protocol (CCP) as the means by which the end-user may request access to a desired program (see DAVIC 1.3.1 part 7). The SDB-CCP is executed between the end-user and a network entity known as the *Broadcast Control Unit (BCU).* In order to support the SDB-CCP the Broadcast Control Unit shouldhave knowledge of service information relevant to broadcast services available at the Replication Unit. In most SDV networks the Broadcast Control Unit and the Replication Unit are a component within the Access Network.

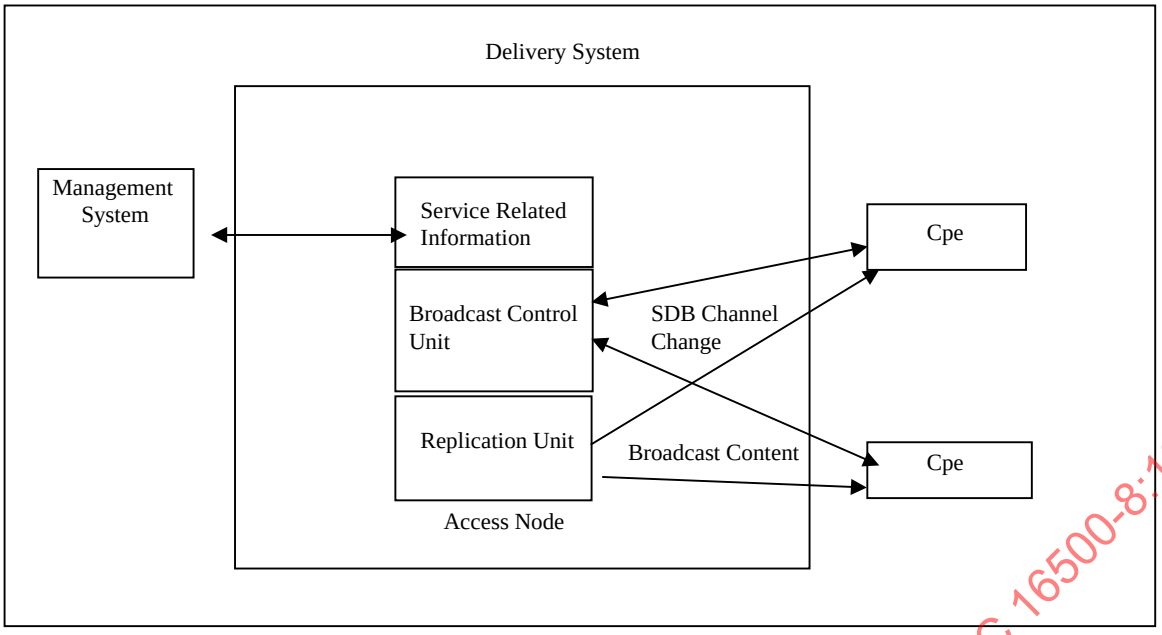


Figure 8-9 — SDV model for switched broadcast services

8.2.3.1.1 SDB Channel Change Protocol

The SDB Channel Change protocol is described in ISO/IEC 16500-5. A brief summary of this protocol is provided here to motivate aspects of the service model. Note that in the following section we refer to the Cpe as the STU to be consistent with DAVIC's terminology.

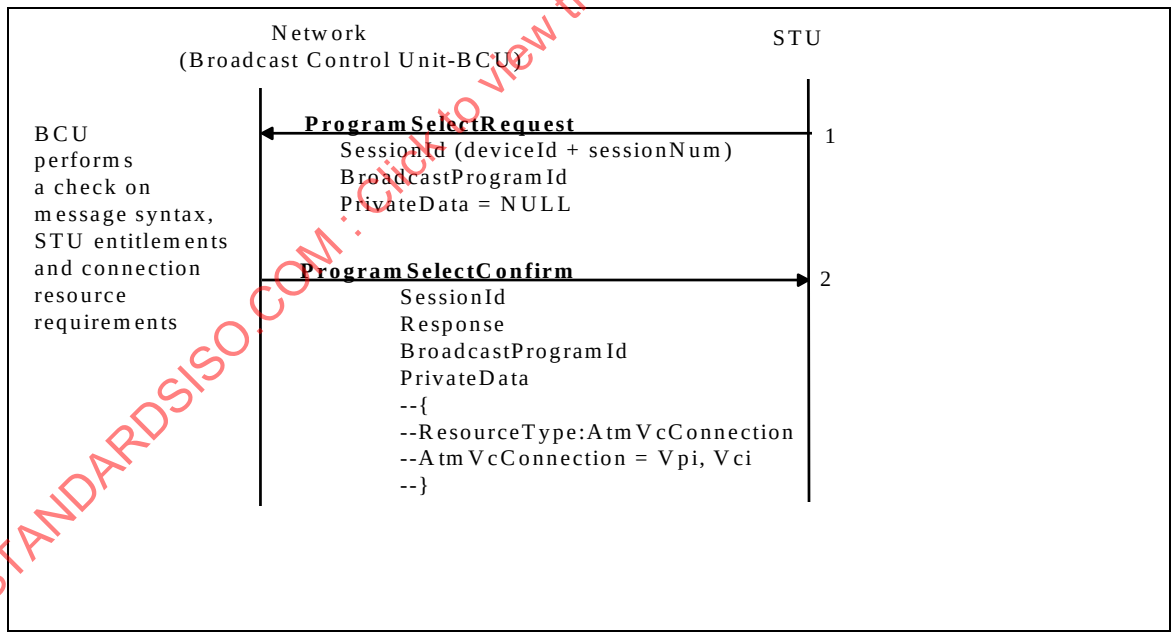


Figure 8-10 — User-Initiated Channel Change Protocol

In Step 1, the STU (End-User) sends a ProgramSelectRequest message to the BCU. The message contains the BroadcastProgramId, which is a network-wide unique handle that identifies a particular broadcast program channel. The SessionId serves to uniquely identify a particular session associated with a given STU. The SessionId is 10 bytes in length. DAVIC specifies that the first 6 bytes of the SessionId correspond to the

deviceIdId (serial number) of the STU. The last 4 bytes correspond to the Session Number that serves to distinguish between multiple sessions associated with a given STU. Typically, a single session is used for broadcast services and a default session value is assumed.

In Step 2, the BCU shoulddetermine whether the STU is entitled to receive the requested broadcast program and whether the connection resources are available to meet the request. If the STU is entitled and the network can support the request, the BCU responds with the ProgramSelectConfirm message with the response code of rspOK, and provides the connection resource associated with the BroadcastProgramId within the private data field. In the case of an FTTC network, the connection resource is an AtmVcConnection, however in other networks alternate connection resource descriptors may also be used. The desired broadcast program stream is then switched to from the Replication Unit to the STU. If the STU is not entitled to the requested program the response code within the ProgramSelectConfirm message may be set to rspEntitlementFailure and the BCU may provide the Vpi, Vci associated with an alternate “Barker” channel.

A network initiated version of the Channel Change protocol is also supported.

8.2.3.1.2 Information Requirements for the Broadcast Control Unit

The discussion in the previous section illustrates the type of information needed by the Broadcast Control Unit to support SDV broadcast services. To summarize, the BCU shouldhave knowledge of :

- A list of broadcast channels (identified by the BroadcastProgramIds) supported by the network and their associated connection resources available at the Replication Unit. Optionally, a connection resource corresponding to an alternative broadcast channel (Barker) may also be specified for each valid broadcast channel.
- A list of CPE that are allowed to access services from the network.
- The entitlements associated with each CPE.

8.2.3.1.3 Entitlement Categories

A broadcast service typically offers broadcast programs in the category of basic, premium and pay-per-view channels. All subscribers with a service account are entitled to basic programming. Entitlement to premium channels requires a special subscription and the subscription profile varies from subscriber to subscriber. Similarly, pay-per-view events shouldbe purchased by the subscriber to obtain access. Pay-per-view events require entitlements to a channel, which are time-dependent. Other time-dependent entitlements relate to free-previews and promotions. These entitlements however, are not subscriber-specific but apply to the general subscriber base in a given service area. Hence, these entitlements may be considered as network-wide entitlements applicable to a particular broadcast channel.

8.2.4 Service Management View

The need to manage Channel Change and Session Management in the Delivery System motivates the definition of most of the managed objects in this model. One possible exception is the service provider managed object which may not be required in the Delivery System, but will be needed at the Service Management Layer (SML).

The model provides an integrated view of service management for the Delivery System. However, some attributes are not required if the managing system is only managing the Access Node (BCU) or only the SRM. The model explicitly identifies which objects and attributes would be instantiated at the Access Node (BCU). Conversely, if an attribute is needed for BCU management, but may not be needed for service management layer functionality, the model indicates that the attribute is optional or not required for the Service Management Layer (SML). Hence, from the point of view of the Broadcast Control/Replication Unit only a subset of the service related model specified for the Delivery System is required, and the elements that are *not* appropriate are clearly indicated within the GDMO model.

8.3 Information Model Overview

This protocol-independent information model is provided to aid reader’s understanding of the data involved in this management interface. The objects correspond to the managed objects in the CMISE information model in subclause 8.4.

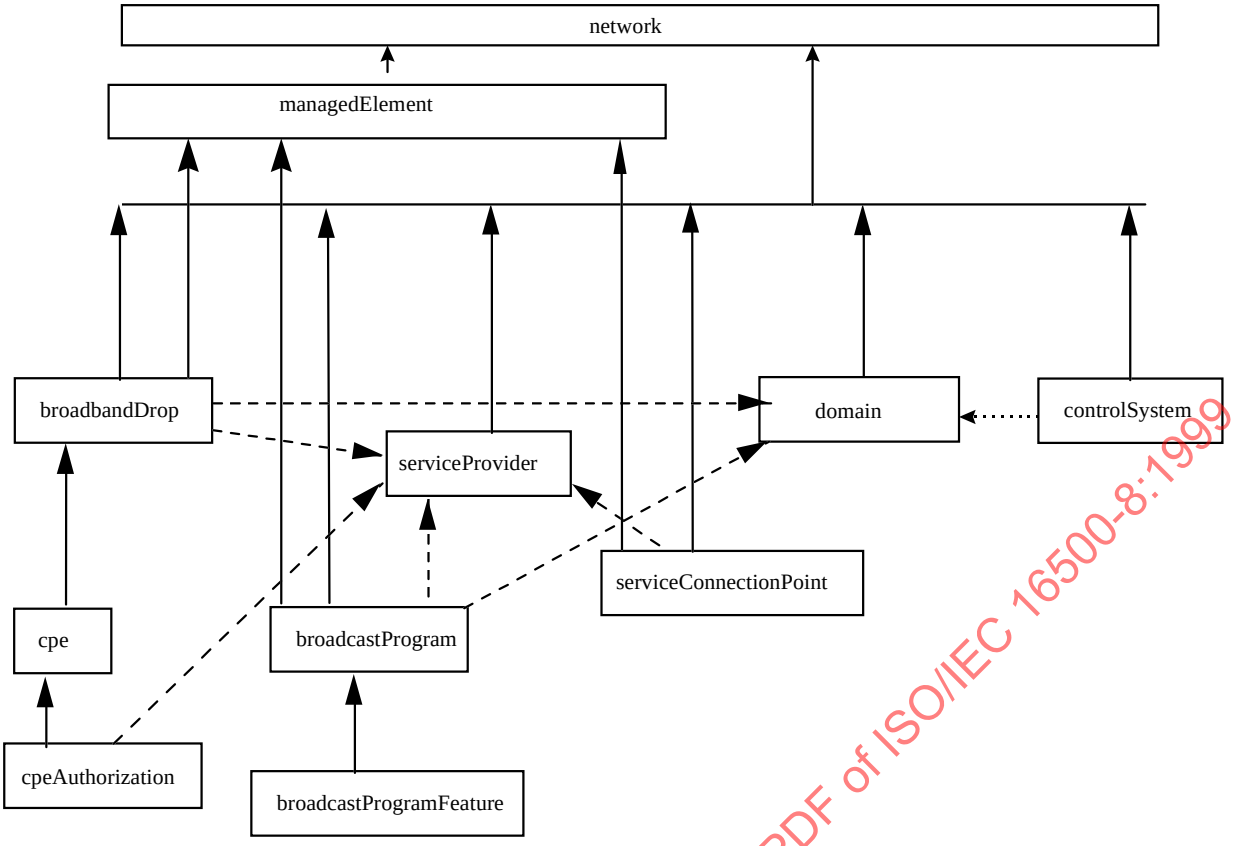


Figure 8-11 — Naming and Relationship Diagram for Information Model

8.3.1 broadbandDrop

This object serves as the network point of attachment for the CPE. The broadbandDrop object must be supported at the BCU.

- ***broadbandDropId,*** is the network address or other globally identifier for the broadbandDrop, such as the E-164 address. This attribute shall be maintained at the BCU/Replication Unit and the SML and shall be the naming attribute for this managed object.
- ***broadbandDropLocalIndex,*** is an optional local identifier for a subscriber drop. This attribute is required at the BCU but not in the SML.
- ***broadbandDropPhysicalAddress***, provides the physical addressing for a subscriber drop. In the case of a FTTC network, this attribute will indicate OnuId and PortId. This attribute is required at the BCU but not at the SML.
- ***administrativeState***, indicates whether the drop is enabled or disabled (locked or unlocked). This attribute is required at the BCU but not at the SML.
- ***operationalState***, indicates whether the drop is in-service. This attribute is required at the BCU but not at the SML.
- ***pointerToDomain***, identifies the service area for this broadband drop. This attribute is optional at the BCU and required at the SML.
- ***subscription***, identifies what services a subscriber is authorized for. This attribute is a set of serviceProvider, serviceType, serviceSubTypes. This attribute is optional at the BCU and required at the SML.

8.3.2 broadcastProgram

Represents a network channel from a service provider service gateway to the access network. It is used for digital broadcast and pay-per-view services. There is one instance for each network channel that a service provider has

in a network.. An instance of this object is created when a new broadcast channel is created. This object is required at the BCU/Replication Unit and SML.

- ***broadcastProgramId*** is a globally unique identifier for a broadcast channel within a network. This attribute must be maintained at the BCU.
- ***programConnectionIdentifier*** serves as an identifier for the connection resource required to support a broadcast channel. For an ATM network, this attribute indicates the port, Vpi, Vci values. This attribute must be maintained at the BCU.
- ***barkerConnectionIdentifier*** serves as an identifier for the connection resource of the "barker channel" (which may be provided at the CPE should the CPE's request for a particular broadcastProgram fail an entitlement check). Support for a barkerChannel is optional, at the BCU as well as at the Service Management Layer.
- ***channelType*** indicates the type of the broadcastProgram (premium or pay-per-view). This attribute is optional at the BCU.
- ***domains*** identify the serving areas for a broadcast channel. This attribute is optional for the BCU.
- ***broadcastProgramServiceProvider*** indicates the service provider for this broadcast channel. This attribute is optional at the BCU.
- ***administrativeState*** is an optional attribute used to identify if the broadcastProgram is enabled or disabled. This attribute will be maintained at the BCU.
- ***operationalState*** is an optional attribute used to identify whether a broadcastProgram is in-service or out-of-service. This attribute will be maintained at the BCU.

8.3.3 broadcastProgramFeature

This object identifies events that have been defined by a service provider to run on a broadcast channel. There are a variety of different event types, which may be provided, such as pay-per-view content, promotions, freeView. This data is used to allocate resources in the Delivery System to provide the event to authorized subscribers. Often this authorization is provided to all subscribers in a broadcast area. At the appointed time, the Delivery System is updated to allocate the appropriate network resources to provide the program event to the designated subscribers. After the event concludes, the network resources in the Delivery System are de-allocated.

This object is only instantiated at the BCU for broadcastProgramFeatures of type "promotion" or "freeView." A pay-per-view event will not be visible at the BCU unless it is purchased by a cpe. There is one instance of this class for each event for each broadcast program. This object will be name-bound to the broadcastProgram object.

As an example of the use of the attributes to implement broadcast program features, the promotion feature allows cpe in the broadcast area unlimited access to an unsubscribed broadcast channel. At the Service Management Layer, the Service Provider may specify a beginDt (endDt and interval are null) for the promotion. However, at the BCU the event will be instantiated exactly when the event is to begin. Therefore, at the BCU the beginDt is also null. When the Service Provider wishes the promotion to end, the corresponding managed object at the SML is deleted and the BCU object is also immediately deleted.

Another kind of special offer, sometimes call dynamic free view (or preview), allows a cpe to request access to an unsubscribed channel for a specified duration during a promotional period. For example, a cpe can view channel 1 for 2 hours up to 3 times starting November 11 at 7 p.m. and ending November 12 at 7 p.m. In this case, at the SML, the Service Provider will specify a startDt and duration, 2 hours is the interval and the freeViewCount is 3.

As an example of the use of attributes to specify freeView feature parameters, a cpe may be authorized to view channel 1 for 4 minutes up to 3 times per hour. In this case, 4 minutes is the interval and 3 times is the freeViewCount. The startTime identifies when the freeView should become available. The freeView remains available until canceled by the service provider; the endDate attribute is null for a freeView event.

- ***broadcastProgramFeatureId***, uniquely identifies the broadcast program feature on a broadcast channel. This unique identifier is created by concatenating several fields: the broadcast program feature type, the broadcast program feature start time, the broadcast program feature duration. The final component required to uniquely identify this broadcast program feature is the broadcastProgramId which is the parent object for this managed object; this component is not repeated in the concatenated broadcastProgramFeatureId field.
  - The ***Feature Type***, indicates the type of the feature (pay-per-view, promotion, freeView, etc.). Only a subset of the allowed types will be supported at the BCU. For instance, the pay-per-view type will not be supported.

  - The ***Start Time***, indicates the begin-date and time for this event.
  - The ***Feature Duration,*** indicates the length of the event in seconds.
- ***freeViewCount***, indicates the number of times a freeView broadcast program event can be displayed within an interval. This attribute is optional as it only applies to features of type freeView.
- ***interval***, identifies the repeat interval for freeView events. This attribute is optional as it only applies to features of type freeView.
- ***endDt***, identifies the end date and time for when a freeView feature should stop being broadcast. This attribute is optional as it only applies when the service provider has defined a definite end date and time for the promotion or freeView feature.
- ***timeZone,*** identifies the offset from Greenwich Mean Time for the startTime of the broadcastProgramFeature. This attribute is optional and is not required at the BCU.

8.3.4 controlSystem

This object class contains information identifying the systems used in the network provider domain that manage the services, including servers which manage interactive service connections and pay-per-view service connections. There is one instance for each control system type within a network. This object is not needed at the BCU but is needed at the SML. Attributes of this object may also be required at the DIIP server which is typically instantiated in the Access Node.

- ***controlSystemType***, identifies the type of control system, such as pay-per-view or interactive video
- ***address,*** identifies the IP address or ATM address of the control system where signaling information from the cpe should be sent
- ***domains,*** identifies the service areas for which the control system has control

8.3.5 cpe

This object is used for information on the customer premise equipment or set top hardware. It maintains the relationship between a cpe logical/physical address and its broadbandDrop. The cpe object is name-bound to the broadbandDrop that supports it. It also identifies what services a cpe is authorized to receive; this data is used by the BCU to control cpe access to broadcast channels. Event-type entitlements are defined in the cpeAuthorization managed object. There is one instance for each cpe device on a subscriber's broadband drop. The cpe object is required at the BCU/Replication Unit. Attributes of this object may also be required at the DIIP server which is typically implemented in the Access Node.

- ***cpeID,*** is the logical identifier for the cpe. This attribute will be maintained at the BCU.
- ***cpeType***, describes the cpe device (e.g. vendor, model, boot software version). This attribute is optional at the BCU.
- ***cpeStatus,*** describes whether a cpe is authenticated or unauthenticated. A cpe, which has undergone UN-Configuration, is considered authenticated. This attribute is optional at the BCU/Replication Unit.
- ***cpeEntitlements***, identifies broadcast channels which the cpe is authorized for. This attribute will be maintained at the BCU.
- ***statusDt,*** provides the date when the cpe went into its current status. This attribute is optional for the BCU.
- ***channelIdBoot***, maintains the broadcastProgramId for the boot channel for the set top. This attribute is optional for the BCU.
- ***cpeDeviceId,*** is a physical identifier for the cpe, such as the MAC address. This attribute will be maintained at the BCU.
- ***cpeAddress,*** this attribute identifies the IP address for the cpe. This attribute is not needed at the BCU, but is required at the SML.
- ***administrativeState,*** identifies if a cpe is enabled or disabled. This attribute will be maintained at the BCU.
- ***operationalState***, identifies if the cpe is in-service or out-of-service. This attribute will be maintained at the BCU.

Actions:

- ***sendMsg***, this message is used to send an emergency broadcast message or service assurance message to a cpe.

Actions are not required when this object is supported in the BCU/Replication Unit.

8.3.6 cpeAuthorization

This object grants a cpe access to a broadcast program for a finite duration of time. There is one instance of this class for each pay-per-view request for each cpe. The object class is name-bound to the cpe to which this authorization applies. The object must be supported at the BCU.

- ***cpeAuthorizationId,*** is a unique identifier for an authorization which is a concatenation of several fields: broadcastProgramId; cpeAuthorizationStartTime; cpeAuthorizationDuration. The concatenated fields are:
  - ***broadcastProgramId***, identifies the broadcastProgram for which access is granted.
  - ***cpeAuthorizationStartTime***, identifies the time when the access to a program takes effect.
  - ***cpeAuthorizationDuration***, defines the duration of the granted access.
- ***timeZone,*** identifies the offset from Greenwich Mean Time when this cpeAuthorization should begin. This attribute is optional and is not needed at the BCU.
- ***cpeAuthorizationServiceProvider***, identifies the service provider who submitted the authorization for this cpe. This attribute is optional and is not needed at the BCU; it is needed at the SML.

8.3.7 domain

This object is used to maintain reference information about the distribution areas for broadband services. It is used to identify what areas are served by a particular broadcast channel, as well as by what areas a particular subscriber is served. There is one object for each instance of a serving area. This object is not supported in the BCU/Replication Unit.

- domainId
- domainName

8.3.8 serviceConnectionPoint

Represents information about the signaling connection used by a service provider server element to communicate with a network provider for particular services. The connection can be a virtual channel connection (VCC) between the service provider server and the network provider via an ATM switch. This object is not supported by the BCU/Replication Unit, but is required at the SML.

- ***serviceConnectionPointId***, uniquely identifies the video provider service. This will be the E.164 address of the service.
- ***address***, identifies the network address where the service provider signaling information is sent during session connection. This will be an IP address for the service.
- ***pointerToServiceProvider,*** identifies the serviceProvider for this signaling connection service.
- ***serviceStatus,*** identifies whether this service is active or disabled.

8.3.9 serviceProvider

There is one instance of this object for each service provider. This data identifies whether the Service Provider is still active. This object is not supported at the BCU/Replication Unit, but may be required at the SML.

- ***serviceProviderId***, uniquely identifies a service provider.
- ***serviceProviderStatus***, indicates whether new customers can be accepted from this service provider.

8.4 GDMO Information Model

Managed object classes are provided in subclause 4.1, followed by Conditional Packages in subclause 4.2, Attributes in subclause 4.3, Naming Bindings in subclause 4.4, and ASN.1 syntax in subclause 4.5. This information model conforms to the specification format defined in the Guidelines for the Definition of Managed Objects (GDMO) specified in ITU-T X.722 | ISO/IEC 10165-4.

8.4.1 Managed Object Classes

8.4.1.1 broadbandDrop

```
broadbandDrop MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY broadbandDropPkg PACKAGE
    BEHAVIOUR broadbandDropBeh;
    ATTRIBUTES
      broadbandDropId
        SET-BY-CREATE-GET
        ;;;
CONDITIONAL PACKAGES
    subscriptionPkg
      PRESENT IF "This object is used at the SML to manage the services associated with a broadband drop.
This
                  package is not instantiated at the BCU/Replication Unit.",
    broadbandDropBcuPkg
      PRESENT IF "This object is instantiated at the BCU/Replication Unit.";
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlmanagedObjectClass(3) 1};

broadbandDropBeh BEHAVIOUR
  DEFINED AS
    "The broadbandDrop managed object class contains attributes about the subscriber's drop. The
broadbandDropId uniquely identifies the broadbandDrop and could be the E.164 for the drop. This object
contains a conditional package that identifies user subscriptions represented by a pointer to a service provider, a
service type, and a service subtype. To update the subscriptions for a subscriber, the managing system will add or
remove values from the set of services in the subscription attribute. This object also contains a conditional
package of identifiers instantiated at the BCU, but not needed at the SML. This conditional package includes a
broadbandDropPhysicalAddress which represents the physical equipment connections for the drop, a
broadbandDropLocalIndex which is a locally assigned value at the BCU which is used to simplify data access,
and administrative/operational state attributes.";
```

8.4.1.2 broadcastProgram

```
broadcastProgram MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY broadcastProgramPkg PACKAGE
    BEHAVIOUR broadcastProgramBeh;
    ATTRIBUTES
      broadcastProgramId
        SET-BY-CREATE-GET,
      programConnectionIdentifier
        SET-BY-CREATE-GET;;;
CONDITIONAL PACKAGES
    barkerConnectionIdentifierPkg
      PRESENT IF "The object supports identification of a default channel to which the cpe should be
connected if
                  the cpe is not authorized for the requested broadcast program/channel.",
    broadcastProgramServiceManagementPkg
      PRESENT IF "This object is used for service management of the relationship between broadcast programs,
                  service providers, and domains. This package is not present at the BCU.",
    "ITU-T Recommendation M.3100":administrativeOperationalStatesPackage
      PRESENT IF "This object is instantiated at the BCU and manages administrative and operational states.";
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlmanagedObjectClass(3) 2};
```


broadcastProgramBeh BEHAVIOUR
DEFINED AS
"This object class contains information identifying the particular physical broadcast channels being delivered by a service provider. It will be instantiated per physical broadcast channel and represents a network channel from a service provider service gateway to the access network. It is used for digital broadcast and pay-per-view services. There is one instance for each network channel that a service provider has in an access network. An instance of this object is created when a new broadcast channel is created. This object is needed at the BCU/Replication Unit.";

8.4.1.3 broadcastProgramFeature

broadcastProgramFeature MANAGED OBJECT CLASS
DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
CHARACTERIZED BY broadcastProgramFeaturePkg PACKAGE
BEHAVIOUR broadcastProgramFeatureBeh;
ATTRIBUTES
broadcastProgramFeatureId
SET-BY-CREATE-GET;;;
CONDITIONAL PACKAGES
repeatProgramPkg
PRESENT IF “the broadcastProgramFeatureType is freeView.",
freeViewEndPkg
PRESENT IF "the broadcastProgramFeatureType is freeView and an endDate has been specified.",
timeZonePkg
PRESENT IF "this object is instantiated at the SML and the time zone is required.";
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlmanagedObjectClass(3) 3};
broadcastProgramFeatureBeh BEHAVIOUR
DEFINED AS
"This object identifies events that have been defined by a service provider to run on a broadcast channel. There are a variety of different event types which may be provided, such as pay-per-view content, promotions, freeView. This data is used to allocate resources in the Delivery System to provide the event to authorized subscribers. Often this authorization is provided to all subscribers in a broadcast area. At the appointed time, the Delivery System is updated to allocate the appropriate network resources to provide the program event to the designated subscribers. After the event concludes, the network resources in the Delivery System are de-allocated. This object is only instantiated at the BCU for broadcastProgramFeatures of type 'promotion' or 'freeView.' A pay-per-view event will not be visible at the BCU unless it is purchased by a cpe. There is one instance of this class for each event for each broadcast program. This object will be name-bound to the broadcastProgram object. As an example of the use of the attributes to implement broadcast program features, the promotion feature allows cpe in the broadcast area unlimited access to an unsubscribed broadcast channel. At the Service Management Layer, the Service Provider may specify a beginDt (endDt and interval are null) for the promotion. However, at the BCU the event will be instantiated exactly when the event is to begin. Therefore, at the BCU the beginDt is also null. When the Service Provider wishes the promotion to end, the corresponding managed object at the SML is deleted and the BCU object is also immediately deleted. Another kind of special offer, sometimes called dynamic free view (or preview), allows a cpe to request access to an unsubscribed channel for a specified duration during a promotional period. For example, a cpe can view channel 1 for 2 hours up to 3 times starting November 11 at 7 p.m. and ending November 12 at 7 p.m. In this case, at the SML, the Service Provider will specify a startDt and endDt, 2 hours is the interval and the freeViewCount is 3. As an example of the use of attributes to specify freeView feature parameters, a cpe may be authorized to view channel 1 for 4 minutes up to 3 times per hour. In this case, 4 minutes is the interval and 3 times is the freeViewCount. The startTime identifies when the freeView should become available. The freeView remains available until canceled by the service provider; the endDate attribute is null for a freeView event. The Feature Id uniquely identifies the broadcast program feature on a broadcast channel. This unique identifier is created by concatenating several fields: the broadcast program feature type, the broadcast program feature start time, the broadcast program feature duration. The final component required to uniquely identify this broadcast program feature is the broadcastProgramId which is the parent object for this managed object; this component is not repeated in the concatenated broadcastProgramFeatureId field. The Feature Type, indicates the type of the feature (pay-per-view, promotion, freeView, etc.). (Only a subset of the allowed types will be supported at the BCU. For instance, the pay-per-view type will not be supported.) The Start Time, indicates the begin date and time for this event. The Feature

Duration, indicates the length of the event in seconds. Several optional attributes are also included in conditional packages which are used for freeView type features. The freeViewCount, indicates the number of times a freeView broadcast program event can be displayed within an interval. The interval, identifies the repeat interval for freeView events. This attribute is optional as it only applies to features of type freeView. The endDt, identifies the end date and time for when a freeView feature should stop being broadcast. This attribute is optional as it only applies when the service provider has defined a definite end date and time for the promotion or freeView feature. The timeZone, identifies the offset from Greenwich Mean Time for the startTime of the broadcastProgramFeature. This attribute is optional and is not required at the BCU.";

8.4.1.4 controlSystem

```
controlSystem MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec.X.721(1992)| ISO/IEC 10165-2:1992":system;
  CHARACTERIZED BY controlSystemPkg PACKAGE
    BEHAVIOUR controlSystemBeh;
    ATTRIBUTES
      controlSystemType
          GET-REPLACE,
      address
          GET-REPLACE,
      domains
          GET-REPLACE
          ADD-REMOVE;;;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlmanagedObjectClass(3) 4};

controlSystemBeh BEHAVIOUR
  DEFINED AS
    "This object class contains information identifying the control systems that manage Session connection and
service authorization. The domains attribute identifies the service areas for which a control systems is valid. The
address identifies the signaling address for signaling between the cpe and the control system. This object will be
instantiated during system configuration per control system. This object is not present in the Access Network
(BCU), but is part of the SML function in the Delivery System.";
```

8.4.1.5 cpe

```
cpe MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY cpePkg PACKAGE
    BEHAVIOUR cpeBeh;
    ATTRIBUTES
      cpeId
          SET-BY-CREATE-GET,
      cpeEntitlements
          GET-REPLACE
          ADD-REMOVE,
      cpeDeviceId
          SET-BY-CREATE-GET;
    ACTIONS
      sendMsg;;;
  CONDITIONAL PACKAGES
    cpeManagementPkg
      PRESENT IF "The object is instantiated at the SML and supports management of cpe status, cpe IP
address,
                  and boot channel identifier. This package is not present at the BCU.",
    cpeBcuPkg
      PRESENT IF "The object is instantiated at the BCU and supports administrative and operational state
                  management.";
```


REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlmanagedObjectClass(3) 5};

cpeBeh BEHAVIOUR
  DEFINED AS
    "This managed object contains information about a subscriber customer premise equipment or set-top-box. An instance of a cpe object exists for every service termination device attached to a broadband drop.";

8.4.1.6 cpeAuthorization

cpeAuthorization MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY cpeAuthorizationPkg PACKAGE
    BEHAVIOUR cpeAuthorizationBeh;
    ATTRIBUTES
      cpeAuthorizationId
        SET-BY-CREATE-GET;;;
  CONDITIONAL PACKAGES
    cpeAuthorizationServiceManagementPkg
      PRESENT IF "This object is instantiated at the SML.";
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlmanagedObjectClass(3) 6};

cpeAuthorizationBeh BEHAVIOUR
  DEFINED AS
    "An instance of this object grants a cpe access to a broadcast program for a finite duration of time, such as when a pay-per-view event has been purchased by the cpe. At the Service Management Layer and at the BCU the cpeAuthorizationId serves as an identifier for this object.";

8.4.1.7 domain

domain MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY domainPkg PACKAGE
    BEHAVIOUR domainBeh;
    ATTRIBUTES
      domainId
        GET,
      domainName
        GET-REPLACE;;;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlmanagedObjectClass(3) 7};

domainBeh BEHAVIOUR
  DEFINED AS
    "This object class contains service management information identifying the particular domains that are defined for broadband services. This managed object is not instantiated in the Access Network/BCU but may be required in the SML.";

8.4.1.8 serviceConnectionPoint

serviceConnectionPoint MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY serviceConnectionPointPkg PACKAGE
    BEHAVIOUR serviceConnectionPointBeh;
    ATTRIBUTES
      serviceConnectionPointId
        GET,
      address

```
        GET-REPLACE,
    pointerToServiceProvider
        GET-REPLACE,
    serviceStatus
        GET-REPLACE;;;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlmanagedObjectClass(3) 8};

serviceConnectionPointBeh BEHAVIOUR
  DEFINED AS
    "This object class contains identification information for a service provider server which communicates with a
Session Manager for signaling information. This object class is instantiated in the Delivery System to support
Session Management.";
```

8.4.1.9 serviceProvider

```
serviceProvider MANAGED OBJECT CLASS
  DERIVED FROM "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":top;
  CHARACTERIZED BY serviceProviderPkg PACKAGE
    BEHAVIOUR serviceProviderBeh;
    ATTRIBUTES
      serviceProviderId
        GET,
      serviceProviderStatus
        GET-REPLACE;;;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlmanagedObjectClass(3) 9};

serviceProviderBeh BEHAVIOUR
  DEFINED AS
    "This object class contains information about a service provider. It will be instantiated at the Service
Management Layer per Video Information Provider.";
```

8.4.2 Conditional Packages

8.4.2.1 barkerConnectionIdentifierPkg

```
barkerConnectionIdentifierPkg PACKAGE
  BEHAVIOUR barkerConnectionIdentifierBeh;
  ATTRIBUTES
    barkerConnectionIdentifier
      SET-BY-CREATE-GET;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 1};
```

8.4.2.2 broadbandDropBcuPkg

```
broadbandDropBcuPkg PACKAGE
  BEHAVIOUR  broadbandDropBcuPkgBeh;
  ATTRIBUTES
    broadbandDropLocalIndex
      SET-BY-CREATE-GET,
    broadbandDropPhysicalAddress
      SET-BY-CREATE-GET,
    "X.721":administrativeState
      GET-REPLACE,
    "X.721":operationalState
      GET;
```


REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlpackage(4) 2};

broadbandDropBcuPkgBeh BEHAVIOUR
  DEFINED AS
    "This package contains attributes used at the BCU for managing the broadbandDrop. The broadbandDropLocalIndex is a local unique identifier used to simplify data retrieval at the BCU.";

8.4.2.3 broadcastProgramServiceManagementPkg

broadcastProgramServiceManagementPkg PACKAGE
  BEHAVIOUR broadcastProgramServiceManagementPkgBeh;
  ATTRIBUTES
    channelType
      GET-REPLACE,
    broadcastProgramServiceProvider
      GET-REPLACE,
    domains
      GET-REPLACE
      ADD-REMOVE;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlpackage(4) 3};

broadcastProgramServiceManagementPkgBeh BEHAVIOUR
  DEFINED AS
    "This package contains attributes used in the Service Management Layer for managing broadcast programs from a network view.";

8.4.2.4 cpeBcuPkg

cpeBcuPkg PACKAGE
  BEHAVIOUR cpeBcuPkgBeh;
  ATTRIBUTES
    "Recommendation X.721:1992":operationalState
      GET,
    "Recommendation X.721":administrativeState
      GET-REPLACE;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlpackage(4) 4};

cpeBcuPkgBeh BEHAVIOUR
  DEFINED AS
    “This package contains administrative and operational state attributes for management at the BCU.";

8.4.2.5 cpeManagementPkg

cpeManagementPkg PACKAGE
  BEHAVIOUR cpeManagementPkgBeh;
  ATTRIBUTES
    cpeType
      GET-REPLACE,
    channelIdBoot
      GET-REPLACE,
    cpeStatus
      GET-REPLACE,
    statusDt
      GET,
    cpeAddress
      GET-REPLACE;
  ACTIONS

```
    sendMsg;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 5};

cpeManagementPkgBeh BEHAVIOUR
  DEFINED AS
    "This package contains attributes used for management of the CPE which are instantiated in the SML of the
Delivery System.";
```

8.4.2.6 cpeAuthorizationServiceManagementPkg

```
cpeAuthorizationServiceManagementPkg PACKAGE
  BEHAVIOUR cpeAuthorizationServiceManagementPkgBeh;
  ATTRIBUTES
    timeZone
        SET-BY-CREATE-GET,
    cpeAuthorizationServiceProvider
        SET-BY-CREATE-GET;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 6};

cpeAuthorizationServiceManagementPkgBeh BEHAVIOUR
  DEFINED AS
    "This package identifies the type of authorization and the Service Provider who submitted the authorization.
This package will not be used at the BCU, but may be instantiated at the SML.";
```

8.4.2.7 freeViewEndPkg

```
freeViewEndPkg PACKAGE
  BEHAVIOUR freeViewEndPkgBeh;
  ATTRIBUTES
    endDt
        SET-BY-CREATE-GET;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 7};

freeViewEndPkgBeh BEHAVIOUR
  DEFINED AS
    "Identifies the end date for freeView programs.";
```

8.4.2.8 repeatProgramPkg

```
repeatProgramPkg PACKAGE
  BEHAVIOUR repeatProgramPkgBeh;
  ATTRIBUTES
    freeViewCount
        SET-BY-CREATE-GET,
    interval
        SET-BY-CREATE-GET;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 8};

repeatProgramPkgBeh BEHAVIOUR
  DEFINED AS
    "Contains information only applicable to service events of type=freeView. Identifies the number of times a
particular broadcastProgramFeature can be viewed by a subscriber and the interval between freeView events.";
```

8.4.2.9 subscriptionPkg

```
subscriptionPkg PACKAGE
```



```
  BEHAVIOUR subscriptionPkgBeh;
  ATTRIBUTES
    pointerToDomain
        GET-REPLACE,
    subscription
        GET-REPLACE
        ADD-REMOVE;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 9};

subscriptionPkgBeh BEHAVIOUR
  DEFINED AS
    "Contains information about the services authorized at a broadband drop (e.g., interactive TV, video on
demand) and the service area where a broadband drop is served.";
```

8.4.2.10 timeZonePkg

```
timeZonePkg PACKAGE
  BEHAVIOUR timeZonePkgBeh;
  ATTRIBUTES
    timeZone
        SET-BY-CREATE-GET;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlpackage(4) 10};

timeZonePkgBeh BEHAVIOUR
  DEFINED AS
    "This package identifies the offset from Greenwich mean time for an event.";
```

8.4.3 Attributes

8.4.3.1 address

```
address ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.Address;
  MATCHES FOR EQUALITY;
  BEHAVIOUR addressBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 1};

addressBeh BEHAVIOUR
  DEFINED AS
    "This attribute identifies a network address. It can be the address for a cpe or server. For the cpe object it
identifies an IP or E.164 address for the cpe. For the serviceProvider object it identifies where the service
provider signaling information should be sent when a subscriber selects that service from the Service Gateway.
For the controlSystem object it identifies the port address where signaling information should be sent for a control
system.";
```

8.4.3.2 barkerConnectionIdentifier

```
barkerConnectionIdentifier ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ConnectionIdentifier;
  MATCHES FOR EQUALITY;
  BEHAVIOUR barkerConnectionIdentifierBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 2};

barkerConnectionIdentifierBeh BEHAVIOUR
  DEFINED AS
    "This attributes serves as a link to the connection resource provisioned to carry the Barker channel.";
```

8.4.3.3 broadbandDropLocalIndex

broadbandDropLocalIndex ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadbandDropLocalIndex;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadbandDropLocalIndexBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 3};

broadbandDropLocalIndexBeh BEHAVIOUR
  DEFINED AS
    "This attribute is a local identifier for the broadbandDrop, included at the BCU to facilitate implementation.";

8.4.3.4 broadbandDropId

broadbandDropId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadbandDropId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadbandDropIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 4};

broadbandDropIdBeh BEHAVIOUR
  DEFINED AS
    "This attribute is a network-wide unique address of the broadband drop and contains addressing information. This could be the E.164 address of the drop.";

8.4.3.5 broadbandDropPhysicalAddress

broadbandDropPhysicalAddress ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadbandDropPhysicalAddress;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadbandDropPhysicalAddressBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 5};

broadbandDropPhysicalAddressBeh BEHAVIOUR
  DEFINED AS
    "The broadbandDropPhysicalAddress serves to locate the drop within the network. For FTTC and VDSL architectures this field represents the ONU Id and Port Id. For ADSL architectures the ONU Id may not be used. This attribute must be specified when creating the drop at the BCU.";

8.4.3.6 broadcastProgramFeatureId

broadcastProgramFeatureId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadcastProgramFeatureId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadcastProgramFeatureIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 6};

broadcastProgramFeatureIdBeh BEHAVIOUR
  DEFINED AS
    "This attribute serves as the naming attribute for the broadcastProgramFeature object. This is a composite field containing the following data as described in the broadcastProgramFeature managed object: broadcast Program Id, begin date, broadcast Program Feature Type, and broadcast Program Feature Duration. This field along with the name-binding of this object to the broadcastProgram object uniquely identifies an instance of this object at the BCU and at the Service Management Layer.";

8.4.3.7 broadcastProgramId

```
broadcastProgramId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadcastProgramId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadcastProgramIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 7};

broadcastProgramIdBeh BEHAVIOUR
  DEFINED AS
    "This is a network-wide unique identifier for a broadcast channel.";
```

8.4.3.8 broadcastProgramServiceProvider

```
broadcastProgramServiceProvider ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.BroadcastProgramServiceProvider;
  MATCHES FOR EQUALITY;
  BEHAVIOUR broadcastProgramServiceProviderBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 8};

broadcastProgramServiceProviderBeh BEHAVIOUR
  DEFINED AS
    "Identifies the service provider for a broadcast program.";
```

8.4.3.9 channelIdBoot

```
channelIdBoot ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.PointerToBroadcastProgram;
  MATCHES FOR EQUALITY;
  BEHAVIOUR channelIdBootBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 9};

channelIdBootBeh BEHAVIOUR
  DEFINED AS
    "The channel identifier for the boot channel for a subscriber set top box. This field should include the NSAP
that identifies the default server for the cpe user. This will be the server which supplies initial download or
applications to the cpe. It is used in the serverId field of the networkConfigurationParameters section of the
DSM-CC U-N Configuration Message";
```

8.4.3.10 channelType

```
channelType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ChannelType;
  MATCHES FOR EQUALITY;
  BEHAVIOUR channelTypeBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 10};

channelTypeBeh BEHAVIOUR
  DEFINED AS
    "Identifies a type of channel (e.g., analog, digital broadcast or other).";
```

8.4.3.11 controlSystemType

```
controlSystemType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ControlSystemType;
  MATCHES FOR EQUALITY;
  BEHAVIOUR controlSystemTypeBeh;
```

REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 11};

controlSystemTypeBeh BEHAVIOUR
  DEFINED AS
    "Identifies a type of control system (e.g. pay per view, interactive video). An initial choice of values is defined for this attribute, but new values may also be defined.";

8.4.3.12 cpeAddress

cpeAddress ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.Address;
  MATCHES FOR EQUALITY;
  BEHAVIOUR cpeAddressBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 12};

cpeAddressBeh BEHAVIOUR
  DEFINED AS
    "This attribute contains the IP address for a cpe.";

8.4.3.13 cpeAuthorizationId

cpeAuthorizationId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.CpeAuthorizationId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR cpeAuthorizationIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 13};

cpeAuthorizationIdBeh BEHAVIOUR
  DEFINED AS
    "An identifier or handle for the cpeAuthorization object. This attribute serves as the naming attribute for the cpeAuthorization object. This is a composite field consisting of the following data: broadcast program identifier, start time, and cpe Authorization duration. This field along with the name binding of this object to the cpe object serves to uniquely identify the cpeAuthorization object at the BCU and at the Service Management Layer.";

8.4.3.14 cpeAuthorizationServiceProvider

cpeAuthorizationServiceProvider ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.PointerToServiceProvider;
  MATCHES FOR EQUALITY;
  BEHAVIOUR cpeAuthorizationServiceProviderBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 14};

cpeAuthorizationServiceProviderBeh BEHAVIOUR
  DEFINED AS
    "Identifies the service provider who submitted a cpe authorization.";

8.4.3.15 cpeEntitlements

cpeEntitlements ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.CpeEntitlements;
  MATCHES FOR EQUALITY, SUBSTRINGS;
  BEHAVIOUR cpeEntitlementsBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 15};

cpeEntitlementsBeh BEHAVIOUR

DEFINED AS
"Identifies the broadcast channels which the cpe is authorized for.";

8.4.3.16 cpeId

cpeId ATTRIBUTE
WITH ATTRIBUTE SYNTAX AttributeModule.NameType;
MATCHES FOR EQUALITY, ORDERING, SUBSTRINGS;
BEHAVIOUR cpeIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 16};

cpeIdBeh BEHAVIOUR
DEFINED AS
"Identifies a set top unit logical address, or the logical identifier for a CPE. If the printable string choice for syntax is used, then matching on the substrings is permitted. If the numericName choice for the syntax is used, then matching on ordering is permitted.";

8.4.3.17 cpeDeviceId

cpeDeviceId ATTRIBUTE
WITH ATTRIBUTE SYNTAX AttributeModule.CpeDeviceId;
MATCHES FOR EQUALITY;
BEHAVIOUR cpeDeviceIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 17};

cpeDeviceIdBeh BEHAVIOUR
DEFINED AS
"Identifies MAC address or other identifier used for signaling from the cpe. ";

8.4.3.18 cpeStatus

cpeStatus ATTRIBUTE
WITH ATTRIBUTE SYNTAX AttributeModule.CpeStatus;
MATCHES FOR EQUALITY;
BEHAVIOUR cpeStatusBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 18};

cpeStatusBeh BEHAVIOUR
DEFINED AS
"This attribute is an identifier for the status of the cpe (i.e. unauthenticated, authenticated, etc.)";

8.4.3.19 cpeType

cpeType ATTRIBUTE
WITH ATTRIBUTE SYNTAX AttributeModule.CpeType;
MATCHES FOR EQUALITY;
BEHAVIOUR cpeTypeBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 19};

cpeTypeBeh BEHAVIOUR
DEFINED AS
"Identifies the type of cpe used by a subscriber. This information may be used to distinguish between different manufacturers, models, and software releases for subscriber equipment.";

8.4.3.20 domainId

domainId ATTRIBUTE

```
  WITH ATTRIBUTE SYNTAX AttributeModule.DomainId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR domainIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 20};

domainIdBeh BEHAVIOUR
  DEFINED AS
    "This attribute is the domain identifier which uniquely identifies any instance of a geographical domain type.
";
```

8.4.3.21 domainName

```
domainName ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.DomainName;
  MATCHES FOR EQUALITY;
  BEHAVIOUR domainNameBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 21};

domainNameBeh BEHAVIOUR
  DEFINED AS
    "A text description of a domain.";
```

8.4.3.22 domains

```
domains ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.Domains;
  MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
  BEHAVIOUR domainsBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 22};

domainsBeh BEHAVIOUR
  DEFINED AS
    "Identifies the domains to which an entity provides service.";
```

8.4.3.23 endDt

```
endDt ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.EndDate;
  MATCHES FOR EQUALITY;
  BEHAVIOUR endDtBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 23};

endDtBeh BEHAVIOUR
  DEFINED AS
    "The date and time for when a promotional feature should stop being made available to CPEs. This attribute is
used at the Service Management layer to specify the end date for a particular broadcastProgramFeature type such
as promotion.";
```

8.4.3.24 interval

```
interval ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.Duration;
  MATCHES FOR EQUALITY;
  BEHAVIOUR intervalBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 24};
```


intervalBeh BEHAVIOUR
  DEFINED AS
    "This is the time interval for which the freeView access is granted.";

8.4.3.25 pointerToDomain

pointerToDomain ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.PointerToDomain;
  MATCHES FOR EQUALITY;
  BEHAVIOUR pointerToDomainBeh;
REGISTERED AS{ iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 25};

pointerToDomainBeh BEHAVIOUR
  DEFINED AS
    "Identifies the domain that a video drop is connected to.";

8.4.3.26 pointerToServiceProvider

pointerToServiceProvider ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.PointerToServiceProvider;
  MATCHES FOR EQUALITY;
  BEHAVIOUR pointerToServiceProviderBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 26};

pointerToServiceProviderBeh BEHAVIOUR
  DEFINED AS
    "Identifies the service provider for an entity.";

8.4.3.27 programConnectionIdentifier

programConnectionIdentifier ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ConnectionIdentifier;
  MATCHES FOR EQUALITY;
  BEHAVIOUR programConnectionIdentifierBeh;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 27};

programConnectionIdentifierBeh BEHAVIOUR
  DEFINED AS
    "Identifies the underlying connection for a broadcast program. This could be the port Id on an atm switch and vpi/vci or (for HFC networks) HFC frequency and mpeg Program Id. ";

8.4.3.28 serviceConnectionPointId

serviceConnectionPointId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ServiceConnectionPointId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR serviceConnectionPointIdBeh;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 28};

serviceConnectionPointIdBeh BEHAVIOUR
  DEFINED AS
    "Uniquely identifies a service connection point. This may be a combination of a serviceId (e.g. E.164 address) and a serviceType (e.g., PPV, IMTV). As an implementation option, the serviceType part may be null.";

8.4.3.29 serviceProviderId

```
serviceProviderId ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ServiceProviderId;
  MATCHES FOR EQUALITY;
  BEHAVIOUR serviceProviderIdBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 29};

serviceProviderIdBeh BEHAVIOUR
  DEFINED AS
    "This attribute is the identifier of a service provider.";
```

8.4.3.30 serviceProviderStatus

```
serviceProviderStatus ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ServiceProviderStatus;
  MATCHES FOR EQUALITY;
  BEHAVIOUR serviceProviderStatusBeh;
REGISTERED AS {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 30};

serviceProviderStatusBeh BEHAVIOUR
  DEFINED AS
    "Indicates whether new subscribers, pay-per-view events, authorizations, etc. may be accepted from this
service provider.";
```

8.4.3.31 serviceStatus

```
serviceStatus ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.ServiceStatus;
  MATCHES FOR EQUALITY;
  BEHAVIOUR serviceStatusBeh;
REGISTERED AS{ iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 31};

serviceStatusBeh BEHAVIOUR
  DEFINED AS
    "The status of a service provider service. Can be active or disabled.";
```

8.4.3.32 statusDt

```
statusDt ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.StatusDt;
  MATCHES FOR EQUALITY;
  BEHAVIOUR statusDtBeh;
REGISTERED AS{ iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlattribute(7) 32};

statusDtBeh BEHAVIOUR
  DEFINED AS
    "This attribute is the date when a cpe went into the status that it is currently in.";
```

8.4.3.33 subscription

```
subscription ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.Subscription;
  MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
  BEHAVIOUR subscriptionBeh;
```


REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 33};

subscriptionBeh BEHAVIOUR
  DEFINED AS
    "Identifies the subscription parameters which apply to an entire broadband drop, e.g. whole house flag (whether the whole house should receive a program if a single CPE has been authorized for it.";

8.4.3.34 freeViewCount

freeViewCount ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.FreeViewCount;
  MATCHES FOR EQUALITY;
  BEHAVIOUR freeViewCountBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 34};

freeViewCountBeh BEHAVIOUR
  DEFINED AS
    "A count indicating the number of times a subscriber can see a freeView on a channel during a particular period of time. This may or may not be supported on the access technology where the subscriber is connected.";

8.4.3.35 timeZone

timeZone ATTRIBUTE
  WITH ATTRIBUTE SYNTAX AttributeModule.TimeZone;
  MATCHES FOR EQUALITY;
  BEHAVIOUR timeZoneBeh;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlattribute(7) 35};

timeZoneBeh BEHAVIOUR
  DEFINED AS
    "The offset from Greenwich Mean Time for an event.";

8.4.4 Name Bindings

8.4.4.1 broadbandDrop-managedElement

broadbandDrop-managedElement NAME BINDING
  SUBORDINATE OBJECT CLASS broadbandDrop AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":
  managedElement AND SUBCLASSES;
  WITH ATTRIBUTE broadbandDropId;
  BEHAVIOUR broadbandDrop-managedElementBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493) davicServiceRelatedControlnameBinding(6) 1};

broadbandDrop-managedElementBeh BEHAVIOUR
  DEFINED AS
    "This name binding represents the physical drop of a customer in a particular access network.";

8.4.4.2 broadbandDrop-network

broadbandDrop-network NAME BINDING

```
  SUBORDINATE OBJECT CLASS broadbandDrop AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":network AND
  SUBCLASSES;
  WITH ATTRIBUTE broadbandDropId;
  BEHAVIOUR broadbandDrop-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 2};

broadbandDrop-networkBeh BEHAVIOUR
  DEFINED AS
    "The broadbandDrop managed object class is instantiated per drop for a Subscriber.";
```

8.4.4.3 broadcastProgram-managedElement

```
broadcastProgram-managedElement NAME BINDING
  SUBORDINATE OBJECT CLASS broadcastProgram AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation. M.3100":
  managedElement AND SUBCLASSES;
  WITH ATTRIBUTE broadcastProgramId;
  BEHAVIOUR broadcastProgram-managedElementBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 3};

broadcastProgram-managedElementBeh BEHAVIOUR
  DEFINED AS
    "The broadcastProgram managed object may be instantiated for each broadcast channel within an access node
represented by a managedElement managed object.";
```

8.4.4.4 broadcastProgram-network

```
broadcastProgram-network NAME BINDING
  SUBORDINATE OBJECT CLASS broadcastProgram AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100”:network AND
  SUBCLASSES;
  WITH ATTRIBUTE broadcastProgramId;
  BEHAVIOUR broadcastProgram-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 4};

broadcastProgram-networkBeh BEHAVIOUR
  DEFINED AS
    "The broadcastProgram managed object is instantiated for each broadcast channel within a network.";
```

8.4.4.5 broadcastProgramFeature-broadcastProgram

```
broadcastProgramFeature-broadcastProgram NAME BINDING
```



```
  SUBORDINATE OBJECT CLASS broadcastProgramFeature AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS broadcastProgram AND SUBCLASSES;
  WITH ATTRIBUTE broadcastProgramFeatureId;
  BEHAVIOUR broadcastProgramFeature-broadcastProgramBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 5};

broadcastProgramFeature-broadcastProgramBeh BEHAVIOUR
  DEFINED AS
    "The broadcastProgramFeature managed object is instantiated per broadcastProgram that has special features
defined. The broadcastProgramFeatureId is a composite field which includes the broadcast program feature start
date, broadcast program feature duration (in seconds), and broadcast program feature type (e.g., promotion,
freeView).";
```

8.4.4.6 controlSystem-network

```
controlSystem-network NAME BINDING
  SUBORDINATE OBJECT CLASS controlSystem AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100":network AND
  SUBCLASSES;
  WITH ATTRIBUTE "CCITT Rec. X.721 (1992) | ISO/IEC 10165-2 : 1992":
  systemId;
  BEHAVIOUR controlSystem-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 6};

controlSystem-networkBeh BEHAVIOUR
  DEFINED AS
    "The control system managed object is instantiated during system configuration.";
```

8.4.4.7 cpe-broadbandDrop

```
cpe-broadbandDrop NAME BINDING
  SUBORDINATE OBJECT CLASS cpe AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS broadbandDrop AND SUBCLASSES;
  WITH ATTRIBUTE cpeId;
  BEHAVIOUR cpe-broadbandDropBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 7};

cpe-broadbandDropBeh BEHAVIOUR
  DEFINED AS
    "The cpe managed object is instantiated per Subscriber per Set- Top-Box.";
```

8.4.4.8 cpeAuthorization-cpe

```
cpeAuthorization-cpe NAME BINDING
  SUBORDINATE OBJECT CLASS cpeAuthorization AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS cpe AND SUBCLASSES;
  WITH ATTRIBUTE cpeAuthorizationId;
  BEHAVIOUR cpeAuthorization-cpeBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    DELETES-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 8};

cpeAuthorization-cpeBeh BEHAVIOUR
  DEFINED AS
    "Identifies the services authorized for a cpe, including broadcast channels, pay-per-view, and pay-per-view
authorizations. The cpeAuthorizationId is a composite naming attribute consisting of the broadcastProgramId,
cpeAuthorizationStartTime and cpeAuthorizationDuration";
```

8.4.4.9 domain-network

```
domain-network NAME BINDING
  SUBORDINATE OBJECT CLASS domain AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS “ITU Recommendation M.3100”:network AND
  SUBCLASSES;
  WITH ATTRIBUTE domainId;
  BEHAVIOUR domain-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 9};

domain-networkBeh BEHAVIOUR
  DEFINED AS
    "The domains may be instantiated per network.";
```

8.4.4.10 serviceConnectionPoint-managedElement

```
serviceConnectionPoint-domain NAME BINDING
  SUBORDINATE OBJECT CLASS serviceConnectionPoint AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS “ITU Recommendation M.3100”:managedElement AND
SUBCLASSES;
  WITH ATTRIBUTE serviceConnectionPointId;
  BEHAVIOUR serviceConnectionPoint-managedElementBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 10};

serviceConnectionPoint-managedElementBeh BEHAVIOUR
  DEFINED AS
    "The serviceConnectionPoint managed object is instantiated per service provider signaling  point  per
domain.";
```


8.4.4.11 serviceConnectionPoint-network

```
serviceConnectionPoint-network NAME BINDING
  SUBORDINATE OBJECT CLASS serviceConnectionPoint AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS network AND SUBCLASSES;
  WITH ATTRIBUTE serviceConnectionPointId;
  BEHAVIOUR serviceConnectionPoint-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 11};

serviceConnectionPoint-networkBeh BEHAVIOUR
  DEFINED AS
    "The serviceConnectionPoint managed object may be instantiated per
    service provider signaling  point  per network.";
```

8.4.4.12 serviceProvider-network

```
serviceProvider-network NAME BINDING
  SUBORDINATE OBJECT CLASS serviceProvider AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS "ITU Recommendation M.3100": network AND
  SUBCLASSES;
  WITH ATTRIBUTE serviceProviderId;
  BEHAVIOUR serviceProvider-networkBeh;
  CREATE
    WITH-REFERENCE-OBJECT,
    WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlnameBinding(6) 12};

serviceProvider-networkBeh BEHAVIOUR
  DEFINED AS
    "The serviceProvider managed object is instantiated per service provider.";
```

8.4.5 Actions

8.4.5.1 sendMsg

```
sendMsg ACTION
  BEHAVIOUR sendMsgBeh;
  MODE CONFIRMED;
  WITH INFORMATION SYNTAX AttributeModule.SendMsg;
  WITH REPLY SYNTAX AttributeModule.ReturnCodeAndMessage;
REGISTERED AS { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlaction(9) 1};

sendMsgBeh BEHAVIOUR
  DEFINED AS
    "This action is used to forward a message to a CPE or to CPEs within a domain.";
```

8.4.6 ASN.1 Module

```
AttributeModule { iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) davic(1493)
davicServiceRelatedControlasn1Module(2) 1}
```

```
DEFINITIONS IMPLICIT TAGS ::=
BEGIN

  --EXPORTS everything

  IMPORTS

  ObjectInstance
  FROM
  CMIP-1 {joint-iso-ccitt(2) ms(9) cmip(1) modules(0) protocol(3)}

  DistinguishedName,
  NameType,
  RelativeDistinguishedName
  FROM
  InformationFramework {joint-iso-ccitt(2) ds(5) modules(1) informationFramework(1)};

  Address ::= SEQUENCE {
        addressType     INTEGER{
        ip              (0),
        atm             (1)},
        addressValue    OCTET STRING(SIZE(1 .. 20))}

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

  BroadbandDropLocalIndex ::= INTEGER (1 .. 65535)

  BroadbandDropPhysicalAddress ::= SEQUENCE {
    portId          PortId,
    onuID           OnuId OPTIONAL} -- Not needed for ADSL access technology

  BroadcastProgramId ::= OCTET STRING(SIZE(4))

  BroadcastProgramFeatureId ::= OCTET STRING(SIZE(14))

  --This is a composite field.  The first 8 bytes 0-7 indicate the broadcast program feature start date.
  --Bytes 8-11 indicate the brodcastProgramFeatureDuration in seconds.

  --Byte 12 indicates the broadcast program feature type. Defined values are: event(0), promotion(1),
  --freeView(2), cancelWindows (3).
  --Byte 13 is reserved.  If optional free-view pkg is included then freeViewCount and freeViewInterval
  --may also be needed in this attribute.  We do not include it presently.
  --The date field is an OCTET STRING of (SIZE(8))
  --The first seven bytes starting from 0 denotes century, year, month, day, hour, minutes, second byte 8 is
reserved.
  --BroadcastProgramFeatureType is presented as an integer with types: event(0), promotion(1),
  --freeView(2), cancelWindow(3),
  --additional types may be defined only promotion and free-view may be needed at the BCU.

  BroadcastProgramServiceProvider ::= ObjectInstance

  ChannelType ::= INTEGER{
    analog              (0),
    digitalBroadcast    (1),
    dataCarousel        (2)}

  ConnectionIdentifier ::= CHOICE {
    atmProgramConnection SEQUENCE {
      portId        PortId,
```



```
    vpi             INTEGER,
    vci             INTEGER},
  hfcProgramConnection SEQUENCE {
    frequency       INTEGER,
    mpegPid         INTEGER}}

ControlSystemType ::= INTEGER{
  payPerView                    (0),
  interactiveVideo              (1),
  config                        (2),
  switchedDigitalBroadcast      (3)}

-- additional types may be defined

CpeAuthorizationId ::= OCTET STRING(SIZE(16))

--This is a composite field.  Bytes 0-3 specify the broadcastProgramId.
--Byte 4-11 are specified according to the syntax associated with the Date attribute and
--indicate the cpe authorization start time.
--Bytes 12-15 specify the cpe Authorization Duration in seconds.
--The date field is an OCTET STRING of size 8.
--The first seven bytes starting from 0 denotes century, year, month, day, hour, minutes, second
--Byte 8 is reserved

CpeEntitlements ::= SET OF BroadcastProgramId

CpeDeviceId ::= OCTET STRING(SIZE(6))

CpeStatus ::= INTEGER{
  authenticated         (0),
  unauthenticated       (1)}

CpeType ::= GraphicString(SIZE(1 .. 8))

DomainId ::= GraphicString(SIZE(1 .. 20))

DomainName ::= GraphicString(SIZE(1 .. 20))

Domains ::= SET OF PointerToDomain

Duration ::= INTEGER(0 .. 99999999)

EncodingType ::= OCTET STRING(SIZE(1))
        -- (values: 01 - null
        -- terminated text, 02 - html text,
        -- 03 - pcm)


FreeViewCount ::= INTEGER(0 .. 99)

MessageAttributeStructure ::= SEQUENCE {
  messageType           MessageType,
  toneFrequency         ToneFrequency,
  beginDate             GeneralizedTime,
  endDate               GeneralizedTime,
  timeZone              TimeZone,
  repeatCount           RepeatCount,
  broadcastProgramId    BroadcastProgramId}
```

```
MessageEncodingStructure ::= SEQUENCE {
  msgLength             MsgLength,
  encodingType          EncodingType,
  msgText               MsgText}

MessageType ::= INTEGER{
  emergency             (0),
  serviceAssurance      (1)}

MsgCode ::= GraphicString(SIZE(9))

MsgLength ::= OCTET STRING(SIZE(2))

MsgText ::= GraphicString

OnuId ::= INTEGER(1 .. 256)

PointerToBroadcastProgram ::= ObjectInstance

PointerToDomain ::= ObjectInstance

PointerToServiceProvider ::= ObjectInstance

PortId ::= INTEGER(1 .. 256)

RepeatCount ::= OCTET STRING(SIZE(1)) -- number of times set top box
--should report the message

ReturnCode ::= OCTET STRING(SIZE(4))

ReturnCodeAndMessage ::= SEQUENCE {
  returnCode                    ReturnCode,
  returnMessageStructure        ReturnMessageStructure}

ReturnMessageStructure ::= SEQUENCE {
  msgCode               MsgCode,
  returnMsgText         ReturnMsgText}

ReturnMsgText ::= GraphicString(SIZE(250))

SendMsg ::= SEQUENCE {
  messageAttributeStructure     MessageAttributeStructure,
  messageEncodingStructure      MessageEncodingStructure}

ServiceConnectionPointId ::= SEQUENCE {
  serviceId     NumericString(SIZE(15)),
  serviceType   INTEGER{
    null (0),
    interactiveTV (1),
    broadcast (2),
    data (3)}}

ServiceProviderId ::= GraphicString(SIZE(1 .. 4))

ServiceProviderStatus ::= INTEGER{
  suspended (0),
  active (1)}
```



```
  ServiceStatus ::= INTEGER{
    disabled (0),
    active (1)}

  ServiceSubType ::= INTEGER{
    wholeHouseFlag (1)}

  ServiceType ::= INTEGER {
          broadcast         (0),
          data              (1) }

StatusDt ::=  GeneralizedTime

  StatusLength ::= OCTET STRING(SIZE(1))

  StatusType ::= OCTET STRING(SIZE(2))

  StatusValue ::= OCTET STRING

  Subscription ::= SET OF SEQUENCE {
    pointToServiceProvider      PointerToServiceProvider,
    serviceType                 ServiceType,
    serviceSubType              ServiceSubType}

Text ::= GraphicString(SIZE(1 .. 40))

  TimeZone ::= INTEGER -- This is a signed integer.

  ToneFrequency ::= OCTET STRING(SIZE(4)) -- duration in seconds for audible tone.


END    -- end of productions
```

Annex A
(informative)
Management protocol independent modeling techniques

A.1 Introduction

The management of the feature rich and highly complex DAVIC systems and applications (evident from the vast range of specifications submitted) requires the assistance of a number of supporting technologies, coherently integrated in a uniform way. Given the network-focused aspect of DAVIC systems management and the above considerations, it has been considered to adopt a pragmatic approach and to base the specification and development of 'DAVIC management systems' on TMN framework and principles.

The full scope of 'service level management' for DAVIC has yet to be addressed. It is likely that contemporary technological developments (e.g. CORBA [OMG96] and TINA [TINA96]) will have an influence on management systems design at the service level, as has been the case in such telecommunication research initiatives as the ACTS program, specifically the ACTS project Prospect [Pros96] [Pros97]. Prospect has found that the design and development of integrated multi-domain service management for the Open Services Market, requires, or at least will significantly benefit from, additional and alternative design methodologies, as compared with the traditional TMN and ODP approaches.

With this in mind, there is a desire to consider more formal notations for use in describing protocol neutral information models and management interfaces.

The consideration of various notations from prominent OOAD techniques and adoption of a particular notation for use in specifying the protocol neutral information models permits a more 'open' interpretation of the specifications. The use of a standard notation whether ad-hoc or industry wide in this task should assist in the incorporation of contemporary interface specification notations, e.g. CORBA IDL, as part of the management systems specifications. Thus, a (DAVIC) standard 'protocol neutral' specification notation paves the way for the use of contemporary technologies and constitutes a first step in providing TMN/CORBA inter-working and integration solutions, where required.

A.1.1 A need for Object-oriented methodologies and notations

As the size and complexity of telecommunication systems increases, the creation of quality software requires considerable investment in resources. Building distributed applications can be difficult and the process is assisted considerably when a design methodology is adopted to support the development process. It is generally agreed that using modular approach to a design problem is an effective way to manage complexity, and this is especially true regarding distributed telecommunications management applications.

In adopting such an approach it is, therefore, necessary that these modules have well-defined interfaces, where relevant abstractions and corresponding behaviors are clearly identified and specified. The quality of a system is influenced in the design by the way its modules are identified and bound together. The use of an object-oriented approach in the design of a system is an effective way of developing this modularity concept. The object-oriented paradigm essentially concerns abstraction and behavior, i.e. identification of objects in the system, their operations (behavior) and their relationships.

Object-oriented modeling and design methodologies represent a particular way of thinking about problems by using models organized around real-world concepts. Object-oriented models are useful for reasoning about and understanding problems, communicating with domain experts, modeling enterprises, and designing systems. The key concept of the object is the combination of both behavior and data within a single conceptual entity. This contrasts with conventional procedural or structural programming which focus on data structures and algorithms. The salient features of object-orientation are those of abstraction, encapsulation, polymorphism and inheritance.

- **identification**: data is quantified into discrete, distinguishable entities called objects,
- **classification**: objects with the same data structure (attributes) and behavior (operations) are grouped into a class,
- **polymorphism**: the same operation may behave differently on different classes,

- **inheritance**: the sharing of attributes and operations among classes based on a hierarchical relationships.

The general process of OOAD is one of identifying the relevant abstractions, their behaviors, relationships, and composition. Then through the application of various analytical mechanisms and procedures the analysis model is refined to evolve a design model. Aspects relating to the creation of an object model will be described later.

A.1.2 Work in Progress at the NMF

The NMF's overall goal in producing solution sets and component sets is that these solution should be usable in a more global context and that it should be possible to integrate these solutions to solve other system design problems. With this in mind, it can be determined that modularity, reuse, extensibility should be among the characteristics of solution/component sets.

The NMF information modeling group has identified that an object-oriented approach to information modeling can add coherency of approach to a specification and increase productivity. Therefore, this group is addressing the use of an object-oriented approach that will:

- Provide a single coherent set of requirements and protocol neutral specifications from which implementation agreements can be produced for specific interface/protocol technologies.
- Enable the use of standard industry tools to assist the production of easy to understand information models.
- Automate the mapping from the protocol neutral specification of the information model to a technology specific definition.
- Encourage reuse of object model definitions.
- Identify reusable elements from object model descriptions, and to establish a repertoire of reusable definitions.

In its undertaking the NMF information modeling group has examined various OOAD tools to assist in the O-O information modeling process and thus, have taken the approach of using the OMT notation in the interim, and to progress to using UML when it deemed stabile. A strategic alliance has been created with Platinum Systems the vendors of Paradigm Plus, whereby automatic translation to target management model, such as, GDMO, SNMP, and CORBA IDL will be provided by the Paradigm design tool.

A.2 Overview of independent description notations

This section provides an overview of two modeling notations which can be used to provide a 'protocol neutral description' of the information models that should accompany the various A8 management interface specifications being submitted to DAVIC. The notations used are those for describing the object model in OMT [RUM 91] and the Unified Modeling Language (UML), which may be viewed as a contemporary derivative of the former.

A.2.1 Overview of OMT and Jacobsen's Use Cases

The Object Modeling Technique is a methodology for modeling systems, capturing the system specification through three separate views. These three views are the

- Object Model : this captures the static structure of the system, that is the object classes, their operations (behavior), and their relationships
- Dynamic Model : this represents the temporal, behavioral, control aspects of the system
- Functional Model: this represents the transformational, 'functional' aspects of the system.

The three models give orthogonal views of the system, which can be represented and manipulated with a uniform notation. Though not completely independent each model can be examined and understood in isolation. As the focus of this annex is directed at information modeling and interface specification, the relevance of the Object and Dynamic Models is further discussed.

The Object Model is the core of an OMT model representing the structural composition of the system, its components and their relationships.

The Dynamic Model shows the time independent behavior of the system, capturing the behavior of objects and the interactions between them. This can be captured using Jacobsen's '*Use Cases*', Use Case scenarios, and message/event trace diagrams.

A Use Case can be thought of as *'A behaviorally related sequence of transactions performed by an actor in a dialogue with the system to provide some measurable value to the actor.'* A Use Case is a primarily user centric analysis for capturing functional requirements focusing on how an external 'body' uses or interacts with the system. They help to ascertain the externally visible behavior of a system. As such they depict actors playing a particular role in the use of a system, as shown in figure A-1. In this we see a course registration collection of use cases. A Use Case description is somewhat ad-hoc, in that it outlines the use of the systems in a free text format.

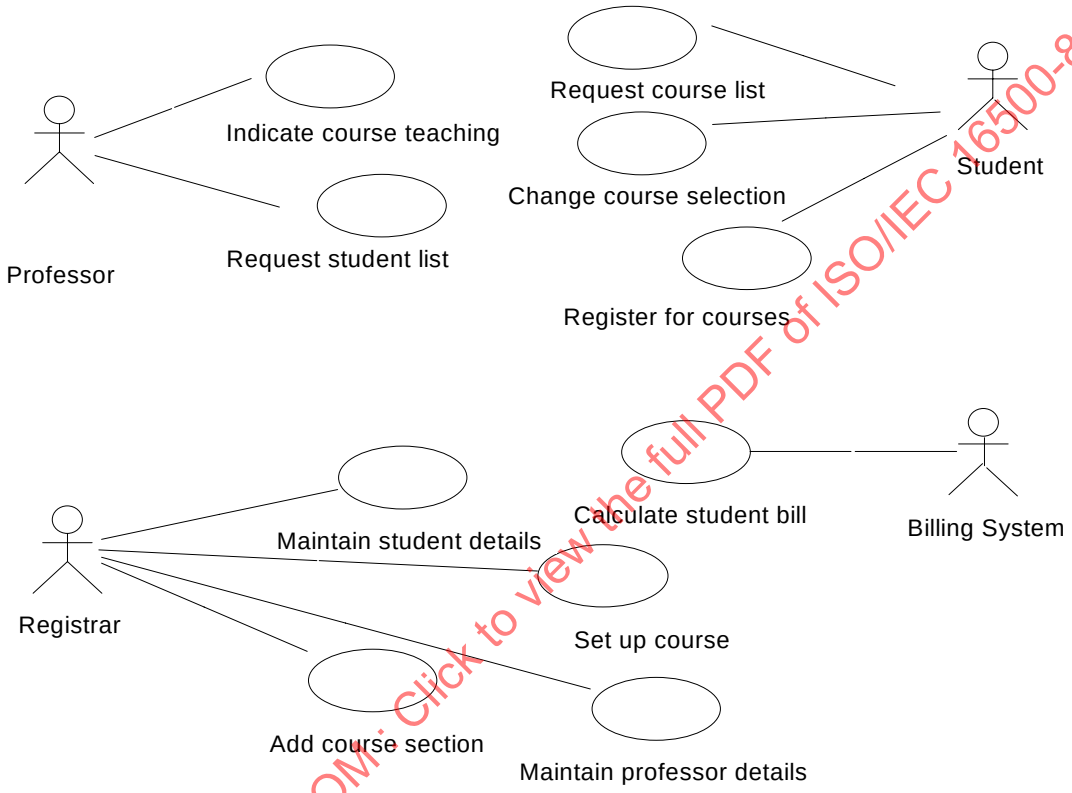


Figure A-1 — Use Case Diagram Depicting Course Management

A Use Case scenario is an instance of a Use Case. These scenarios can be portrayed using 'event trace' or 'message trace' diagrams, which can be helpful when defining the object classes that may be required for a computational model.

The OMT notation is a powerful way of representing Object Models highlighting the classes and the different kinds of relationships between them. The salient symbols of the notation are shown in Figure A-2.

