EBU – Tech 3345

End-to-End IP Network Measurement for Broadcast Applications

Parameters & Management Information Base (MIB)

Geneva
July 2011

Conformance Notation

This document contains both **normative** text and **informative** text.

All text is normative except for that in the Introduction, any section explicitly labelled as ‘Informative’ or individual paragraphs which start with ‘Note:’.

Normative text describes indispensable or mandatory elements. It contains the conformance keywords ‘shall’, ‘should’ or ‘may’, defined as follows:

‘Shall’ and ‘shall not’:	Indicate requirements to be followed strictly and from which no deviation is permitted in order to conform to the document.
‘Should’ and ‘should not’:	Indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others. OR indicate that a certain course of action is preferred but not necessarily required. OR indicate that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.
‘May’ and ‘need not’:	Indicate a course of action permissible within the limits of the document.

Default identifies mandatory (in phrases containing “shall”) or recommended (in phrases containing “should”) presets that can, optionally, be overwritten by user action or supplemented with other options in advanced applications. Mandatory defaults must be supported. The support of recommended defaults is preferred, but not necessarily required.

Informative text is potentially helpful to the user, but it is not indispensable and it does not affect the normative text. Informative text does not contain any conformance keywords.

A conformant implementation is one which includes all mandatory provisions (‘shall’) and, if implemented, all recommended provisions (‘should’) as described. A conformant implementation need not implement optional provisions (‘may’) and need not implement them as described.

Contents

1. Introduction 7

2. Parameters 9

2.1 Scope 9
2.2 General Unit Parameters 9
2.2.1 Power Supply 9
2.2.2 Temperature 10
2.3 Network/Transport Layers 10
2.3.1 Protocols 10
2.3.1.1 Internet Protocol (IPv4) 10
2.3.1.2 Internet Protocol (IPv6) 10
2.3.1.3 Port Number 10
2.3.1.4 Real-time Transport Protocol (RTP) 11
2.3.1.5 Session Initiation Protocol (SIP) 11
2.3.1.6 Internet Group Management Protocol (IGMP) 11
2.3.1.7 Asynchronous Serial Interface (ASI) 12
2.4 Application Layer 12
2.4.1 Audio 12
2.4.2 Audio Signal Format 12
2.4.2.1 ITU G.711 12
2.4.2.2 ITU G.722 13
2.4.2.3 ISO MPEG-1/2 Layer II 13
2.4.2.4 PCM 13
2.4.2.5 ISO MPEG-1/2 Layer III 14
2.4.2.6 MPEG-4 AAC, MPEG-4 AAC-LD 14
2.4.2.7 Enhanced APTx 14
2.4.2.8 MPEG-4 HE-AACv2 15
2.4.2.9 MPEG-4 AAC-ELD 15
2.4.3 Additional Audio Parameters 15
2.4.3.1 Channel Arrangement 15
2.4.3.2 Number of Channels 15
2.4.3.3 Audio Bit Depth 16
2.4.3.4 Sampling Frequency 16
2.4.3.5 Audio Bit Rate 16
2.4.3.6 Audio PID 16
2.4.3.7 Audio Component Number 16
2.4.3.8 Audio Status 17
2.4.3.9 FEC (Forward Error Correction) Type 17
2.4.3.9.1 None 17
2.4.3.9.2 SMPTE 2022/SMPTE 2021-1-2007/RFC2733 17
2.4.3.9.3 Proprietary 17
2.4.3.10 FEC Length & Dimension 18
2.4.4 Video 18
2.4.4.1 Video Status 18

2.4.4.2 Video Signal Format 18
2.4.4.3 Video Source Format 18
2.4.4.3.1 Frame Rate *18*
2.4.4.3.2 Source Type *19*
2.4.4.3.3 Vertical Resolution *19*
2.4.4.3.4 Scan Type *19*
2.4.4.4 Video Coding Type 19
2.4.4.4.1 Uncompressed *19*
2.4.4.4.2 MPEG-2 ..... *19*
2.4.4.4.3 H.264 ..... *20*
2.4.4.4.4 JPEG2000 ..... *20*
2.4.4.4.5 SMPTE VC-2 Video Compression (DIRAC) *20*
2.4.4.4.6 VP8 ..... *20*
2.4.4.4.7 H264 Scalable Extension *20*
2.4.4.5 Video Bit Rate Type 21
2.4.4.6 Video Bit Rate 21
2.4.4.7 Aspect Ratio 21
2.4.4.7.1 Source Aspect Ratio *21*
2.4.4.7.2 Active Format Description *21*
2.4.4.8 FEC (Forward Error Correction) Type 22
2.4.4.8.1 None *22*
2.4.4.8.2 SMPTE 2022 ..... *22*
2.4.4.8.3 Proprietary *22*
2.4.4.9 FEC Length & Dimension 22
2.4.4.10 Trick Mode Support 22
2.4.5 Receiver Network Affected Parameters (Receiver only) 23
2.4.5.1 Buffer Size 23
2.4.5.2 Buffer Occupancy - Time 23
2.4.5.3 Buffer Occupancy - Percentage 23
2.4.5.4 Media Delivery Index (MDI) 23
2.4.5.5 Time Stamped Delay Factor (TS-DF) 24

3. MIB 25

3.1 Rationale 25
3.2 Scope 25
3.3 Overall structure 25
3.3.1 Blocks 25
3.3.2 MIB Trees 26
3.4 Measurement Parameter Details 29
3.4.1 Block Framework MIB objects 29
3.4.1.1 MIB objects for the block framework 29
3.4.1.1.1 blockTable *30*
3.4.1.1.2 blockEntry *30*
3.4.1.1.3 blockId *30*
3.4.1.1.4 blockType *30*
3.4.2 General Unit Parameters 30
3.4.2.1 Power Supply 30
3.4.2.1.1 unitPowerSource *31*

3.4.2.1.2 unitPowerSourceTable 31
3.4.2.1.3 unitPowerSourceEntry 31
3.4.2.1.4 psNumber 31
3.4.2.1.5 psType 31
3.4.2.1.6 psStatus 32
3.4.2.1.7 psChargeLevel 32
3.4.2.1.8 psChargeTime 32
3.4.3 Network/Transport Layers 33
3.4.3.1 Network 33
3.4.3.1.1 nMtBlockTable 33
3.4.3.1.2 nMtBlockEntry 33
3.4.3.1.3 nMtBlockId 33
3.4.3.1.4 nMtIfIndex 33
3.4.3.1.5 nMtTxRxPoint 34
3.4.3.1.6 nMtNetworkType 34
3.4.3.1.7 nMtTransportType 35
3.4.3.1.8 nMtTxRxAddr 35
3.4.3.1.9 nMtPortNumber 35
3.4.3.1.10 nMtIGMPVersion 35
3.4.3.1.11 nMtSIPServerAddr 36
3.4.4 Application Layer 36
3.4.4.1 Audio 36
3.4.4.1.1 aMtBlockTable 36
3.4.4.1.2 aMtBlockEntry 36
3.4.4.1.3 aMtBlockId 37
3.4.4.1.4 aMtAudioComponentNumber 37
3.4.4.1.5 aMtNetworkBlockId 37
3.4.4.1.6 aMtAudioStatus 37
3.4.4.1.7 aMtAudioSignalFormat 38
3.4.4.1.8 aMtAudioPId 38
3.4.4.1.9 aMtIfIndex 38
3.4.4.1.10 aMtFECType 39
3.4.4.1.11 aMtFECLengthDimension 39
3.4.4.2 Video 39
3.4.4.2.1 vMtBlockTable 39
3.4.4.2.2 vMtBlockEntry 40
3.4.4.2.3 vMtBlockId 40
3.4.4.2.4 vMtAudioBlockId 40
3.4.4.2.5 vMtNetworkBlockId 40
3.4.4.2.6 vMtVideoStatus 40
3.4.4.2.7 vMtVideoSourceFormat 40
3.4.4.2.8 vMtVideoCodingType 41
3.4.4.2.9 vMtVideoBitRateType 42
3.4.4.2.10 vMtVideoBitRate 42
3.4.4.2.11 vMtVideoAspectRatio 42
3.4.4.2.12 vMtFECType 43
3.4.4.2.13 vMtFECLengthDimension 43
3.4.4.2.14 vMtTrickModeSupport 43

3.4.4.3 Receiver 43
3.4.4.3.1 rxPointTable *44*
3.4.4.3.2 rxPointEntry *44*
3.4.4.3.3 rxPointBlockId *44*
3.4.4.3.4 rxPointNetworkBlockId *44*
3.4.4.3.5 rxPointBufferSize *45*
3.4.4.3.6 rxPointBufferOcpancyTime *45*
3.4.4.3.7 rxPointBufferOcpncyPcnt *45*
3.4.4.3.8 rxPointMDI *45*
3.4.4.3.9 rxPointTSDF *45*
3.4.4.4 Temperature 46
3.4.4.4.1 temperatureTable *46*
3.4.4.4.2 temperatureEntry *46*
3.4.4.4.3 temperatureBlockId *46*
3.4.4.4.4 temperatureLocnNumber *46*
3.4.4.4.5 temperatureLocation *47*
3.4.4.4.6 temperatureTrend *47*
3.4.4.4.7 temperatureStatus *47*
3.4.4.4.8 temperatureLowWarning *47*
3.4.4.4.9 temperatureHighWarning *48*
3.4.4.4.10 temperatureLowCritical *48*
3.4.4.4.11 temperatureHighCritical *48*

4. References 48

Annex A: (Informative) MIB Definitions from IEC 62379-1 49

Annex B: (Informative) Type and Sequence definitions from IEC 62379-1 51

Annex C: (Informative) Machine-readable measurement block definitions 53

Annex D: (Informative) Machine-readable textual conventions definitions 75

Annex E: (Informative) Machine-readable audio format definitions 79

Annex F: (Informative) Machine-readable video format definitions 105

Annex G: (Informative) Tree of example audio formats 125

Annex H: (Informative) Tree of example video formats 127

End-to-End IP Network Measurement for Broadcast Applications

Parameters & Management Information Base (MIB)

EBU Committee	*First Issued*	*Revised*	*Re-issued*
EC-N	2011		

Keywords: Internet Protocol, Network Management, Broadcasting, MIB

1. Introduction

In recent years, EBU Members have been increasingly adopting IP networks for the contribution of audio and video in real-time. It is well known that although IP networks are of lower cost and provide more flexibility compared with circuit switched networks, they suffer from longer delays and have much larger jitter, while broadcasters' tolerance to these variables is much less than that of normal business IT traffic.

To respond to Members' use of IP the EBU set up two groups, ECN-ACIP (Audio contribution over IP) and ECN-VCIP (Video contribution over IP) with the tasks of drawing up recommended codes of practice[1].

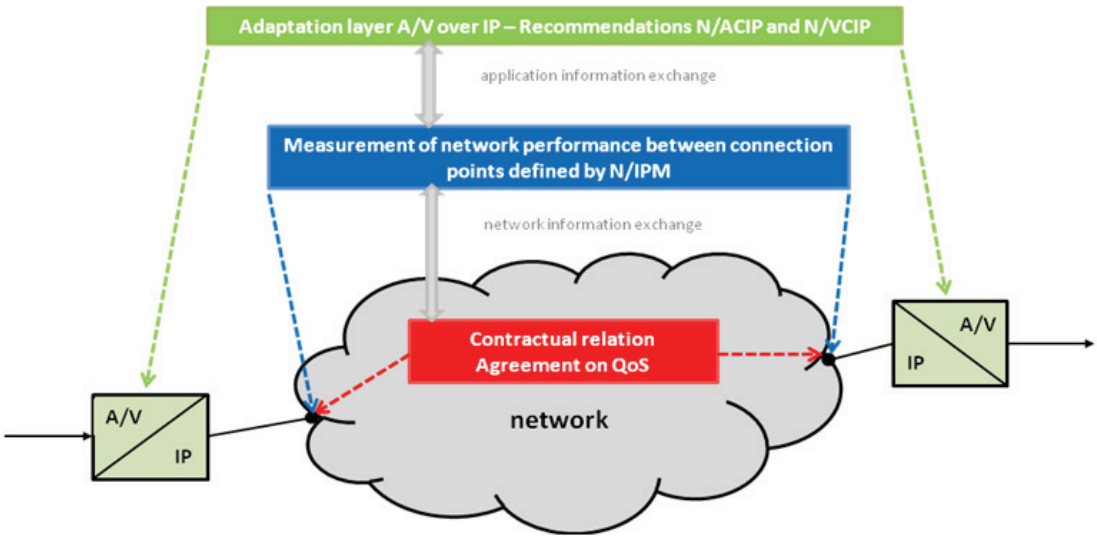


Figure 1: Relationships between ECN groups ACIP, VCIP and IPM

It was also recognised that there would be a strong demand for tools that would enable broadcasters to measure and manage their IP networks properly to suit the many time-critical

[1] ECN-ACIP and ECN-VCIP were formerly known as N/ACIP and N/VCIP respectively.

broadcast applications they would be subjected to. To this end, the ECN-IPM (IP measurement) group was set up. The relationships between these three groups are shown in the above figure.

The goals of the ECN-IPM group were:

- To define a Quality of Service classification to achieve the requested A/V transmission quality for broadcast applications.
- To standardise network information exchange between EBU Members and Network providers.
- To propose a method of collecting end-to-end performance information for management purposes.

In achieving these goals the ECN-IPM Group has specified a set of parameters that are important for broadcasters when using IP networks for audio and video transmission and has developed a software mechanism to probe a network for device and topology discovery, physical path tracing for both end-to-end communication and multicast streams, with the potential for multilayer monitoring for streams on a multi-vendor network with fully media-specific parameters.

The specified parameters cover both the network layer and application layer (for video and audio). SNMP is employed to collect information on the status of networked devices, such as the transmission rate, error rate, the codec used and multicast streams status.

To ensure that all the parameters can be recovered from a variety of different manufacturers' IP equipment, the group has designed a MIB (Management Information Base). Although many MIB files have been published over the years, especially on the network side, very little standardisation work has been done on A/V codec MIB files. The EBU ECN-IPM group has therefore proposed a new standard, based upon IEC 62379 (Common Control Interface for Networked Audio and Video Systems) to address this issue.

Two EBU technical publications have been produced by the ECN-IPM group:

This document, **EBU Tech 3345**, defines the parameters and the new MIB Information.

A description of the software mechanism, **EisStream**[2], may be found in **EBU Tech 3346**. The software is written in Java and it provides physical path tracing for IP traffic using SNMP.

[2] EBU Integrated Monitoring Solution for Media Streams on IP Networks, http://eisstream.sourceforge.net/

2. Parameters

2.1 Scope

This section of the document defines and describes the important parameters required by broadcasters for measurement when using an IP network for audio and video transmission.

The parameters are organised under the following categories:

- **General Unit Parameters**
  - Power Supplies
  - Temperature
- **Network/Transport Layers**
  - Protocols
- **Application Layer**
  - Audio
  - Video
  - Receiver

Each parameter is classified as being mandatory, recommended or optional.

- **Mandatory** indicates that the parameter shall be implemented by all conformant equipment
- **Recommended** indicates that the parameter should be implemented by any conformant equipment
- **Optional** indicates that the parameter may or may not be implemented by any conformant equipment.

2.2 General Unit Parameters

The representation of these parameters in the MIB is shown in § 3.4.2.1 for the power supply and § 3.4.4.4 for temperature.

2.2.1 Power Supply

Class Optional, but mandatory if either multiple and/or chargeable power supplies are present within the unit.

Definition This shall be information about the power supplies for the unit.

Description This shall be the number, type and status of the power supplies supplying power to the unit.

If any of these supplies are battery powered, then information on the charge level and charging time shall also be included.

2.2.2 Temperature

Class Optional, but mandatory if temperature sensors are present within the unit.

Definition This shall be temperature at one or more locations within the unit.

Description This shall be the number and location of temperature sensors within the unit. Each sensor shall provide information about its location and current status, along with levels indicating when warning and critical temperature levels are exceeded.

2.3 Network/Transport Layers

The representation of these parameters in the MIB is shown in § 3.4.3 Network/Transport Layers.

2.3.1 Protocols

2.3.1.1 Internet Protocol (IPv4)

Class Mandatory.

Definition IPv4 is defined in RFC 791, Internet Protocol.

Description The Internet Protocol (IP) is a network layer (Layer 3 in the OSI model) protocol that contains addressing information and some control information to enable packets to be routed in a network. It is the primary network layer in the TCP/IP protocol suite.

IPv4 is the most widely used form of IP.

2.3.1.2 Internet Protocol (IPv6)

Class Optional.

Definition IPv6 is defined in RFC 2460, Internet Protocol, Version 6 (IPv6).

Description Internet Protocol version 6 (IPv6) is the new version of IP based on IPv4. It is a network layer (Layer 3 in the OSI model) protocol that contains addressing information and some control information to enable packets to be routed in a network. It is the primary network layer in the TCP/IP protocol suite.

2.3.1.3 Port Number

Class Mandatory.

Definition A network port is an identification of an application in the TCP or UDP transported datagram. The port number is the number assigned to user sessions and server applications in an IP network.

The port number is related to the definition of a socket as defined in RFC 147. (The combination of IP address and port number is known as a “socket”).

Description The stream port number shall be the number assigned to the port over which the media stream or other application of interest is being measured. (There is a large number of “well-known” port numbers associated with particular protocols). For

example, port 5004 is the recommended port for a media stream using RTP.

2.3.1.4 Real-time Transport Protocol (RTP)

Class Recommended.

Definition RTP is defined in RFC 3550, RTP: A Transport Protocol for Real-Time Applications.

Description Real-time Transport Protocol (RTP) provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. It works on top of UDP.

RTP does not include a loss recovery mechanism in case of packet loss.

2.3.1.5 Session Initiation Protocol (SIP)

Class Mandatory for Audio over IP if SIP is supported and being used, otherwise optional. Refer to EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 4.

Definition SIP is defined in RFC 3261, SIP: Session Initiation Protocol.

Description Session Initiation Protocol (SIP) is an application-layer control (signalling) protocol for creating, modifying, and terminating sessions with one or more participants. These sessions include Internet telephone calls, multimedia distribution, and multimedia conferences.

SIP invitations used to create sessions carry session descriptions that allow participants to agree on a set of compatible media types. SIP makes use of elements called proxy servers to help route requests to the user's current location, authenticate and authorize users for services, implement provider call-routing policies, and provide features to users. SIP also provides a registration function that allows users to upload their current locations for use by proxy servers. SIP runs on top of several different transport protocols.

The particular parameter from SIP shall be the SIP server address to which a unit is registered.

2.3.1.6 Internet Group Management Protocol (IGMP)

Class Mandatory for both version 2 and version 3 if supported, otherwise optional.

Definition IGMP version 2 is defined in RFC 2236, Internet Group Management Protocol, Version 2.

IGMP version 3 is defined in RFC 3376, Internet Group Management Protocol, Version 3.

Description Internet Group Management Protocol (IGMP) is a multicasting protocol used by IP hosts to report their host group memberships to any immediately neighbouring multicast routers.

2.3.1.7 Asynchronous Serial Interface (ASI)

Class Mandatory if supported, otherwise optional.

Definition Although this is not a transport layer under IP, it is still a transportation method. It is listed under the MIB as one of the available selections of TransportType (along with IPv4 and IPv6).

DVB-ASI is specified in EN 50083-9: "Cabled distribution systems for television, sound and interactive multimedia signals; Part 9: Interfaces for CATV/SMATV headends and similar professional equipment for DVB/MPEG-2 transport streams" (DVB Blue Book A010), Annex B, Asynchronous Serial Interface.

Description A digital video broadcast asynchronous serial interface (DVB-ASI) is a serial data transmission protocol used for transporting multi-program transport streams containing audio and video content along with additional associated and un-associated program data.

2.4 Application Layer

2.4.1 Audio

A common set of audio parameters shall be used, whether they are for video devices or audio only devices. However, audio only devices will only use a sub set of these parameters as some of those used for video will not be relevant. For example, most audio only devices may not use PID, whereas this is used within video transport streams.

This list does not include any parameters that although they may affect the audio, are network associated parameters, e.g. transport or signalling protocols, such as RTP, RTCP, SDP, SIP or network impairments, such as packet loss/delay, jitter, etc.

The representation of these parameters in the MIB is shown in § 3.4.4.1 Audio.

2.4.2 Audio Signal Format

Class Mandatory.

Definition The Audio Signal Format shall be defined by the particular format (coding algorithm type or linear) of the audio data in the media stream.

Description The following is a descriptive list of each audio signal format that shall be used.

2.4.2.1 ITU G.711

Class Mandatory.

Definition This is defined in ITU-T Recommendation G.711.

Description The audio data has been encoded using this algorithm as defined in the above recommendation. Two encoding laws are contained in this recommendation, A-Law and μ-Law. Both shall be valid parameters.

G.711 as described in EBU Tech 3326 Audio Contribution over IP, Requirements for

Interoperability, § 3.1.1 shall be supported.

2.4.2.2 ITU G.722

Class Mandatory.

Definition This is defined in ITU-T Recommendation G.722.

Description The audio data has been encoded using this algorithm as defined in the above recommendation using 7 kHz Audio coding within 64 kbit/s.

G.722 as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.1.2 shall be supported.

2.4.2.3 ISO MPEG-1/2 Layer II

Class Mandatory.

Definition MPEG-1 Layer II is defined in ISO/IEC 11172-3.

MPEG-2 Layer II is defined in ISO/IEC 13818-3.

Description These are both lossy audio compression formats supporting a number of sampling and bit rates.

ISO MPEG-1/2 LII as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.1.3 shall be supported.

2.4.2.4 PCM

Class Mandatory.

Definition Apart from these references, no other single definition has been found for PCM.

Alec Harley Reeves, "Electric Signaling System," U.S. Patent 2,272,070, filed November 22, 1939, issued February 3, 1942.

Also French Patent 852,183 issued 1938, and British Patent 538,860 issued 1939.

Description Pulse-code modulation (PCM) is a digital representation of an analogue signal where the magnitude of the signal is sampled regularly at uniform intervals, then quantized to a series of symbols in a numeric (usually binary) code.

PCM as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.1.4 and 3.1.5 shall be supported.

2.4.2.5 ISO MPEG-1/2 Layer III

Class Recommended.

Definition MPEG-1 Layer III is defined in ISO/IEC 11172-3.

MPEG-2 Layer III is defined in ISO/IEC 13818-3.

Description These are both lossy audio compression formats supporting a number of sampling and bit rates.

ISO MPEG-1/2 Layer III as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.2.1 shall be supported, if used.

2.4.2.6 MPEG-4 AAC, MPEG-4 AAC-LD

Class Recommended.

Definition MPEG-4 AAC and MPEG-4 AAC-LD are defined in ISO/IEC 14496-3 and amendments.

Description Advanced Audio Coding (AAC) is a standardised, lossy compression and encoding scheme for digital audio.

MPEG-4 AAC and MPEG-4 AAC-LD as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.2.2 shall be supported, if used.

2.4.2.7 Enhanced APTx

Class Optional.

Definition Enhanced APTx (and variants) is a proprietary algorithm.

Description

- The digital audio data reduction technology known as apt-X is a family of proprietary audio codec compression algorithms developed by APT Licensing. It is based upon time domain Adaptive Differential Pulse Code Modulation (ADPCM).

  Enhanced APTx as described in EBU Tech 3326 Audio Contribution over IP, Requirements for Interoperability, § 3.3.1 shall be supported, if used.

  Agreement on the packetisation of the audio format between the variants of this algorithm is in the process of being effected between the APTx developers and manufacturers, based upon these two documents:
- http://tools.ietf.org/html/draft-gmassey-avt-rtp-aptx-02 from 2008
- http://tools.ietf.org/id/draft-trainor-avt-rtp-aptx-00.txt from 2009

When final agreement has been reached, this and any related documents will be amended.

Note: A new internet draft, titled RTP Payload Format for Standard apt-X and Enhanced apt-X Codecs has recently been filed with the IETF for comment:

https://datatracker.ietf.org/doc/draft-rea-payload-rtp-aptx/ April 2011

2.4.2.8 MPEG-4 HE-AACv2

Class Optional.

Definition MPEG-4 HE-AACv2 (also known as “aacPlus v2”) is defined in ISO/IEC 14496-3 and ISO/IEC 14496-3:2001/Amd.4.

Description Advanced Audio Coding (AAC) is a standardised, lossy compression and encoding scheme for digital audio. MPEG-4 HE-AACv2 is the combination of three technologies:

- Advanced Audio Coding (AAC)
- Spectral Band Replication (SBR)
- Parametric Stereo (PS)

2.4.2.9 MPEG-4 AAC-ELD

Class Optional.

Definition MPEG-4 AAC-ELD is defined in ISO/IEC 14496-3:2005/Amd 9:2008 and ISO/IEC 14496-3:2009

Description Advanced Audio Coding (AAC) is a standardised, lossy compression and encoding scheme for digital audio. AAC-ELD is a combination of MPEG AAC-LD and Spectral Band Replication (SBR).

2.4.3 Additional Audio Parameters

In addition to the audio compression formats above, all the following additional audio parameters are included to match the existing audio signal formats as defined in IEC 62379-2, Annex F.

2.4.3.1 Channel Arrangement

Class Optional.

Definition This is how the individual channels of audio in an audio recording or audio stream are arranged.

Description An audio recording or stream may have one or more channels of audio. This parameter is a description of their arrangement for example, discrete mono, stereo, joint stereo, surround, surround with down mix and so on.

2.4.3.2 Number of Channels

Class Optional.

Definition This is the number of individual channels of audio in an audio recording or audio stream.

Description The number of individual channels of audio, for example, 1 (mono), 2 (stereo or 2 mono channels), and so on.

2.4.3.3 Audio Bit Depth

Class Optional.

Definition The audio bit depth is the number of bits of information recorded for each sample. Bit depth directly corresponds to the resolution of each sample in a set of digital audio data.

Description It is the number of audio bits per sample in the encoded audio signal, such as 16 or 24 bits.

2.4.3.4 Sampling Frequency

Class Mandatory.

Definition The sampling frequency (sampling rate or sample rate) defines the number of samples per second (or per other unit) taken from a continuous signal to make a discrete signal.

Description Sampling Frequency is the number of samples per second (Hz) in the encoded audio signal, usually shown in kHz, such as 44.1 kHz, 48 kHz or 96 kHz.

2.4.3.5 Audio Bit Rate

Class Mandatory.

Definition Audio bit rate is calculated using the following formula:

Bit rate = (bit depth) x (sampling rate) x (number of channels)

Description The audio bit rate represents the amount of information, or detail, which is stored per unit of time of an audio recording.

Transport Stream Related Audio Parameters:

2.4.3.6 Audio PID

Class Mandatory.

Definition Packet ID, used to identify audio/video streams in Digital Video Broadcasting (DVB).

The audio PID defines the data substream containing the Audio (either TV or Radio sound) information.

Description The Programme Identifier for the audio, for example, 0256 or 0327 and so on.

2.4.3.7 Audio Component Number

Class Optional.

Definition The audio component number is the identifier for each audio item, when there are one or more audio items associated with either a single video stream/signal or an

audio only device.

Description The audio component number items may be labelled, for example, “audio 1”, “audio 2”, or similar and so on.

2.4.3.8 Audio Status

Class Mandatory.

Definition Audio status is an indication of the presence, or otherwise, of the audio signal of this audio component number.

Description See definition

2.4.3.9 FEC (Forward Error Correction) Type

Class Mandatory.

Definition FEC Type contains the parameter describing the type of FEC applied to the audio stream.

Description Forward Error Correction (FEC) is a system of error control for data transmission, whereby the sender adds redundant data to its messages, also known as an error-correction code.

The type of FEC applied, if present:

2.4.3.9.1 None

No FEC is being applied.

2.4.3.9.2 SMPTE 2022/SMPTE 2021-1-2007/RFC2733

FEC for use in audio over IP is defined in SMPTE 2022/SMPTE 2021-1-2007/RFC2733 and is a standard for real-time video/audio (RTP) transport over IP networks.

Agreement on which of either the SMPTE standards for FEC or the older RFC2733 (as opposed to RFC 5109 as specified in EBU Tech 3326) is in the process of being agreed between manufacturers and the EBU group ECN-ACIP.

When final agreement has been reached, this and any related documents will be amended.

2.4.3.9.3 Proprietary

A proprietary method of FEC is being applied.

2.4.3.10 FEC Length & Dimension

Class Optional.

Definition FEC Length and Dimension describes the structure of FEC applied to the video stream

Description If FEC is present, then this FEC Length & Dimension parameter shall be required. This shall contain the following information

- Length – an indication of the number of packets over which the FEC is applied
- Dimension – an indication of the matrix size if the FEC is greater than one dimension

For example, Length = 24, Dimension = 5 (to indicate a 5x5 FEC matrix)

2.4.4 Video

The representation of these parameters in the MIB is shown in § 3.4.4.2 Video.

2.4.4.1 Video Status

Class Mandatory.

Definition Video status is an indication of the presence or otherwise, of a particular video stream/signal.

Description See definition

2.4.4.2 Video Signal Format

The Video Signal Format shall be defined by a set of parameters describing the signal format of the video data in the media stream. This shall be either an uncompressed video signal, delivered over a Serial Digital Interface (SDI or HD-SDI) or as compressed video signal coded using a defined coding algorithm type, normally delivered using a Transport Stream.

The following is a descriptive list of each parameter that shall be used to describe the signal format.

2.4.4.3 Video Source Format

The Video Source Format contains a list of basic parameters describing the video signal.

2.4.4.3.1 Frame Rate

Class Mandatory.

Definition The Frame rate of the base video format in Hz

Description See definition

2.4.4.3.2 Source Type

Class Mandatory.

Definition Source Type shall be defined as an indication of the definition of the base video format, such as SD, for example.

Description See definition

2.4.4.3.3 Vertical Resolution

Class Mandatory.

Definition Vertical Resolution shall be defined as the number of lines of vertical resolution used in the base video format.

Description See definition

2.4.4.3.4 Scan Type

Class Mandatory.

Definition Scan Type shall be defined as the type of scanning used in the base video format, such as Progressive, for example.

Description See definition

2.4.4.4 Video Coding Type

The Video Coding Type contains the parameter describing the compression or otherwise of the video signal.

2.4.4.4.1 Uncompressed

Class Mandatory.

Definition The generation of uncompressed video is defined in Recommendation ITU-R BT.601. Refer also to Recommendation ITU-R BT.709 for HD formats.

Description Uncompressed video is video with no coding format (compression) applied.

2.4.4.4.2 MPEG-2

Class Mandatory.

Definition MPEG-2/Video is defined in ISO/IEC 13818-2 and ITU-T Recommendation H.262.

Description MPEG-2 is a standard for "the generic coding of moving pictures and associated audio information".

2.4.4.4.3 H.264

Class Mandatory.

Definition H.264 is defined in Advanced video coding for generic audiovisual services, ITU-T Recommendation H.264 | ISO/IEC International Standard ISO/IEC 14496-10.

Description H.264 is a block-oriented motion-compensation-based standard for video compression.

2.4.4.4.4 JPEG2000

Class Mandatory.

Definition JPEG2000 is defined in ISO/IEC 15444-1:2004 | ITU-T Rec. T.800.

Description JPEG2000 defines a set of lossless (bit-preserving) and lossy compression methods for coding bi-level, continuous-tone grey-scale, palletized colour, or continuous-tone colour digital still images.

2.4.4.4.5 SMPTE VC-2 Video Compression (DIRAC)

Class Mandatory.

Definition SMPTE VC-2 is defined in the proposed FCD SMPTE STANDARD SMPTE 2042-1:200X

Description The VC-2 standard specifies the compressed stream syntax and reference decoder operations for a video compression system. VC-2 is a video compression system utilizing wavelet transforms that decompose the video signal into frequency bands.

VC-2 is a sub-set of a general purpose video codec called “Dirac”.

2.4.4.4.6 VP8

Class Mandatory.

Definition The VP8 Data Format and Decoding Guide is defined in the proposed IETF Internet draft document draft-bankoski-vp8-bitstream-02.

Description VP8 is an open source video compression format developed under the open media project “WebM” by Google.

2.4.4.4.7 H264 Scalable Extension

Class Mandatory.

Definition H.264 Scalable Extension is defined jointly in ITU-T H.264 and ISO/IEC 14496-10:2010. The specification includes that of advanced video coding (AVC), associated extensions to enable scalable video coding (SVC) and multiview video coding (MVC).

The H264 Scalable Extension or Scalable Video Coding (SVC) is contained in Amendment 3 (Annex G) to the standard.

Description H.264/MPEG-4 AVC is a block-oriented motion-compensation-based codec standard for video compression.

2.4.4.5 Video Bit Rate Type

Class Mandatory.

Definition Video Bit Rate Type shall be defined as the type of bit rate used in the base video format, such as variable bit rate (VBR), for example.

Description See definition

2.4.4.6 Video Bit Rate

Class Mandatory.

Definition Video Bit Rate shall be defined as the maximum bit rate of the video stream.

Description The current video bit rate (or video data rate) of the encoded video signal in kilo bits per second (kbit/s) or the maximum if VBR is being used, for example, 1545 kbit/s.

2.4.4.7 Aspect Ratio

The Aspect Ratio contains the list of parameters describing the aspect ratio of the video.

2.4.4.7.1 Source Aspect Ratio

Class Mandatory.

Definition Source Aspect Ratio shall be defined as the aspect ratio of the coded frame as described in the MPEG-2, H264/AVC, or SMPTE VC-1 video syntax. See ITU-T Recommendation H.262 / ISO/IEC 13818-2, ITU-T Recommendation H.264 / ISO/IEC 14496-10:2008 and SMPTE 421M respectively.

Description See definition

2.4.4.7.2 Active Format Description

Class Mandatory.

Definition Active Format Description is defined in SMPTE ST 2016-1:2009

Description The Active Format Description (AFD) is a 4-bit code describing a video picture in terms of the aspect ratio and other characteristics of the active image within the coded frame.

Note: Bar data often used in conjunction with AFD as defined in SMPTE ST 2016-1:2009 and ETSI TS 101 154 V1.9.1 (2009-09) is not currently included in this specification; there will be some restrictions on the types of coded frame that can be reported.

2.4.4.8 FEC (Forward Error Correction) Type

Class Mandatory.

Definition FEC Type contains the parameter describing the type of FEC applied to the video stream.

Description Forward Error Correction (FEC) is a system of error control for data transmission, whereby the sender adds redundant data to its messages, also known as an error-correction code.

The type of FEC applied, if present:

2.4.4.8.1 None

No FEC is being applied.

2.4.4.8.2 SMPTE 2022

FEC for use in video over IP is defined in SMPTE 2022 and is a standard for real-time video/audio (RTP) transport over IP networks.

2.4.4.8.3 Proprietary

A proprietary method of FEC is being applied.

2.4.4.9 FEC Length & Dimension

Class Optional.

Definition FEC Length and Dimension describes the structure of FEC applied to the video stream

Description If FEC is present, then this FEC Length & Dimension parameter shall be required. This shall contain the following information

- Length – an indication of the number of packets over which the FEC is applied
- Dimension – an indication of the matrix size if the FEC is greater than one dimension

For example, Length = 24, Dimension = 5 (to indicate a 5x5 FEC matrix).

2.4.4.10 Trick Mode Support

Class Optional.

Definition Trick Mode Support shall be an indication of the presence, or otherwise of features, such as slow motion or fast forward, supported in the video signal and/or receiver.

Description See definition

2.4.5 Receiver Network Affected Parameters (Receiver only)

The representation of these parameters in the MIB is shown in § 3.4.4.3 Receiver.

2.4.5.1 Buffer Size

Class Mandatory.

Definition This shall be the total receive buffer size, expressed in milliseconds, in the decoding device (receiver). This shall be applicable to both audio only and video devices.

Description See definition

2.4.5.2 Buffer Occupancy - Time

Class Mandatory.

Definition This shall be the quantity of data, expressed in milliseconds, occupying the receive buffer of the decoding device (receiver).

Description See definition

2.4.5.3 Buffer Occupancy - Percentage

Class Mandatory.

Definition This shall be the quantity of data, expressed as a percentage of the total buffer size, occupying the receive buffer of the decoding device (receiver). This shall be applicable to both audio only and video devices.

Description See definition

2.4.5.4 Media Delivery Index (MDI)

Class Optional.

Definition Media Delivery Index (MDI) is defined in RFC 4445.

Description The Media Delivery Index (MDI) is a set of measurements used for monitoring and troubleshooting networks carrying any streaming media type.

The MDI has two components, the Delay Factor (DF) and the Media Loss Rate (MLR). MDI is expressed as simple pair of values in the form of:

DF:MLR

The Delay Factor (DF) is the maximum difference, observed at the end of each media stream packet, between the arrival of media data and the drain of media data.

The Media Loss Rate (MLR) is the number of media packets lost per second. There are two ways to measure it.

- When using MPEG, the sequence numbers of the transport stream can be used. Under error free conditions, the sequence numbers increment. Counting any missing sequence numbers every

second will produce the packets lost per second.

- Alternatively, the layer 4 RTP information can be used as a supplement for the UDP protocol. The RTP has also uses sequence numbers and this number also increments with every packet. The same method as above, of counting missing sequence numbers may be used.

2.4.5.5 Time Stamped Delay Factor (TS-DF)

Class Optional.

Definition A Proposed Time-Stamped Delay Factor (TS-DF) for Measuring Network Jitter with RTP Streams is currently in the process of being presented to the IETF to be assigned the appropriate status and issued with a RFC. Refer otherwise to EBU Tech 3337.

Description It defines a time-stamped delay factor (TS-DF) algorithm that can be used as a tool to measure IP network jitter for applications such as video and audio streaming. It aims to address the measurement problems that the MDI Delay Factor (RFC 4445) has when measuring variable bit rate (VBR) media streams. In constant bit rate (CBR) streams, Ts-DF produces results comparable to the MDI Delay Factor. This algorithm is suitable for measuring IP network jitter in MPEG Transport stream (MPEG TS) over IP and voice over IP, as well as uncompressed stream over IP (such as SDI over IP).

3. MIB

3.1 Rationale

The requirement is to obtain information on the basic parameters of media streams from within manufacturers' equipment connected to a network using SNMP.

To achieve this aim, the existing fundamental control framework of IEC 62379 shall be used for the standardisation and collection of a number of defined parameters associated with media streams, including some network related parameters from manufacturers' current Management Information Bases (MIBs).

3.2 Scope

This section of the document defines and describes the proposed MIB tree and the representation or syntax details of the measurement parameters defined and described in *§ 2 Parameters* of this document.

The parameter ordering will be the same as that in *§ 2 Parameters*.

3.3 Overall structure

The overall structure of this MIB follows that described in IEC 62379-1, where each type of functionality required is based upon a "block" describing that functionality.

3.3.1 Blocks

An item of equipment (a "unit") is regarded as being composed of functional elements or "blocks" which may be linked to each other through internal routing.

Blocks may have inputs, outputs and internal functionality. In general, the output of one block connects to the input of the next block in the processing chain. Blocks can have some associated control parameters and/or status monitoring accessible via the control framework management interface.

There is a special class of blocks called "ports"; ports provide an external connection to other equipment. An "input port" is one where audio, video, or other data enters the unit and an "output port" is one where it leaves the unit.[3].

Within this MIB, another special class of block has been defined; a "measurement" block. This may have zero or more channels on an input connection, but there are no direct outputs or internal functionality. It is merely a means to gather information from within the unit from, for example, other tables and objects or direct from the media stream and for it to be presented and used in a standardised way to a management system. This specification does not define where the information for the content of these blocks is sourced from.

There are five "measurement" information block types defined in this MIB:

[3] Further information on the block structure may be found in IEC 62379-1

- Audio
- Video
- Network
- Receiver
- Temperature[4]

Each of these contains the parameters listed in *§ 2 Parameters* of this document.

In addition, the MIB also includes (imports) some objects associated with the overall block framework control and some standard objects associated with power supplies from the general unit parameters, both from IEC 62379-1.

3.3.2 MIB Trees

The following figures show the MIB files in tree format.

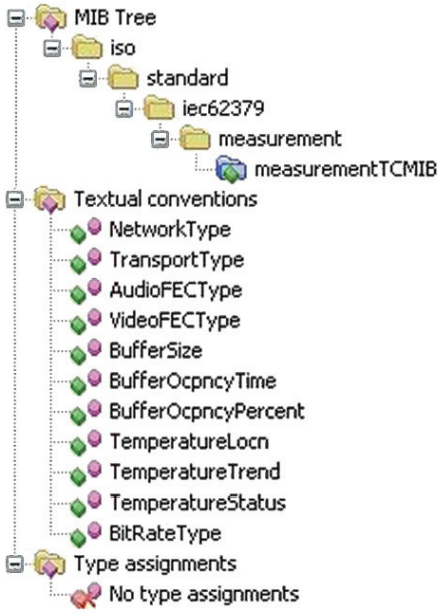


Figure 2: Textual Conventions Module

[4] Although temperature measurement could be considered to be part of a units general parameters, it does not (currently) exist within the defined general unit information objects of IEC 62379-1, so has been defined in [IEC 62379-7] as a separate measurement block

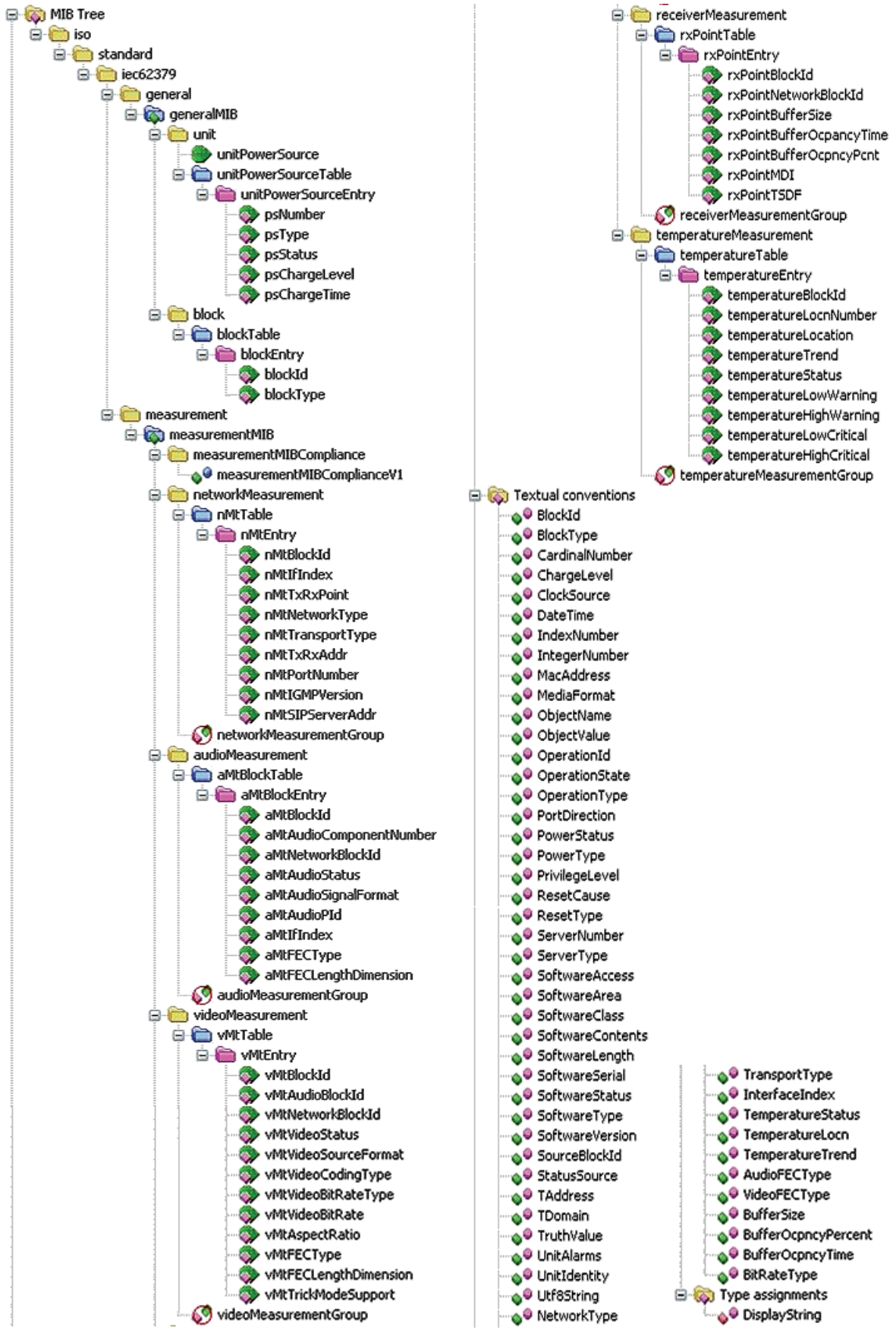


Figure 3: Main Measurement MIB Module

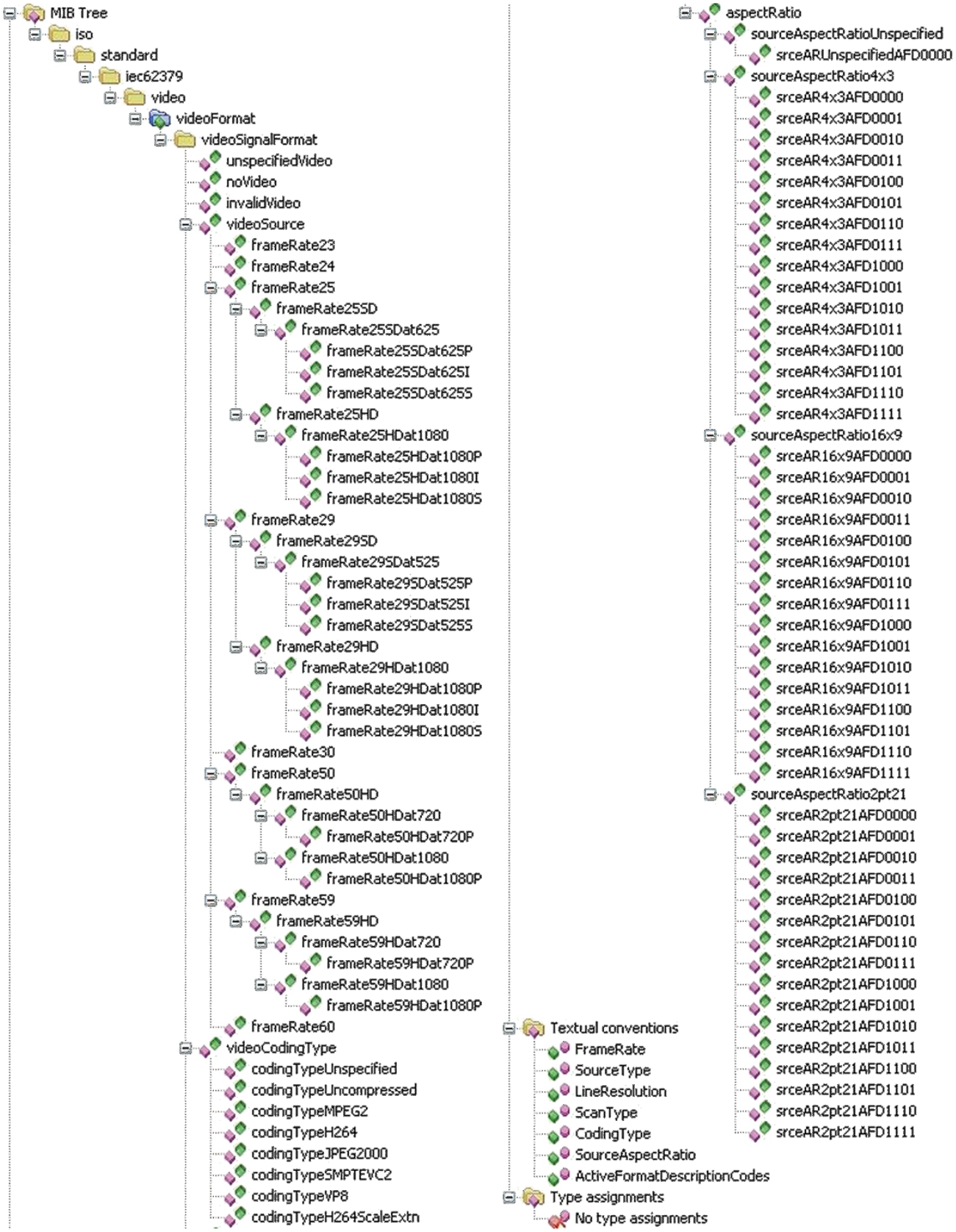


Figure 4: Video Formats Module

Machine readable versions of all these MIB modules are provided for convenience in the various Annexes at the end of this document.

If there is any inconsistency between these machine readable versions and of those in the appropriate sections of:

- IEC 62379-1,
- IEC 62379-2,
- IEC 62379-3 Ed.1 (or later)
- and IEC 62379-7

then

- IEC 62379-1,
- IEC 62379-2,
- IEC 62379-3 Ed.1 (or later)
- and IEC 62379-7

take precedence.

3.4 Measurement Parameter Details

Each of the parameter details lists:

- Parameter Name
- Textual Description
- Any other attributes or conditions

Syntax

3.4.1 Block Framework MIB objects

This section provides a reference to the block framework MIB objects defined in IEC 62379-1 that are used within this standard. If there is any inconsistency between this section and those in the appropriate section of IEC 62379-1, then IEC 62379-1 takes precedence.

Note 1: Only those specific block framework MIB objects used in this standard are included from IEC 62379-1. The remaining objects have been excluded for clarity.

Note 2: Refer to Annex A in this document for a description (extracted from IEC 62379-1) of the various attributes within each of the tables.

3.4.1.1 MIB objects for the block framework

The group of objects in Table 1 shall be implemented by all compliant equipment. The root node for these objects shall be:

{ iso(1) standard(0) IEC62379(62379) general(1) generalMIB(1) block-framework(2) }

Table 1: Managed objects for block and connector configuration

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
blockTable(1)	SEQUENCE OF BlockEntry		none	none	no	m
└blockEntry(1)	BlockEntry		none	none	no	m
├blockId(1)	BlockId	yes	none	none	no	m
└blockType(2)	BlockType		listener	supervisor	no	m

3.4.1.1.1 blockTable

A table of block descriptors. Each block in the unit has an entry in this table.

3.4.1.1.2 blockEntry

An entry in the block table.

3.4.1.1.3 blockId

The block identifier. Used as an index when accessing the block table. Each block in a unit has a unique identifier independent of the block type.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.1.1.4 blockType

The block type identifies the node in the object identifier tree that is the root for any managed objects used to control blocks of this type. Only writable if the unit allows blocks to be reconfigured. Writing the value NULL shall request deletion of the block. Writing a non-NULL value for a non-existent blockId shall request creation of a new block with all its inputs unconnected.

Syntax

BlockType [OBJECT IDENTIFIER]

An object identifier identifying a defined control block.

The block may be one defined in any Part of IEC 62379 or one defined elsewhere.

3.4.2 General Unit Parameters

3.4.2.1 Power Supply

This section provides a reference to the power supply MIB objects defined in IEC 62379-1 that are used within this standard. If there is any inconsistency between this section and those in the appropriate section of IEC 62379-1, then IEC 62379-1 takes precedence.

Note: Only those specific power supply MIB objects used in this standard are included from IEC 62379-1. The remaining objects have been excluded for clarity.

The group of objects in Table 2 shall be implemented by all compliant equipment. The root node for these objects shall be:

{ iso(1) standard(0) IEC62379(62379) general(1) generalMIB(1) unit-information(1) }

Table 2: Managed objects conveying information about the unit

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
unitPowerSource(12)	IndexNumber		listener	supervisor	maybe	m
unitPowerSourceTable(13)	SEQUENCE OF UnitPowerSourceEntry		none	none	no	m
└unitPowerSourceEntry(1)	UnitPowerSourceEntry		none	none	no	m
├psNumber(1)	IndexNumber	yes	none	none	no	m
├psType(2)	PowerType		listener	none	no	m
├psStatus(3)	PowerStatus		listener	none	yes	m
├psChargeLevel(4)	ChargeLevel		listener	none	yes	o
└psChargeTime(5)	CardinalNumber		listener	none	yes	o

3.4.2.1.1 unitPowerSource

The index number of the entry in the unit power source table that represents the current source of power for the unit. Only writable if the unit permits manual switching between power sources.

3.4.2.1.2 unitPowerSourceTable

A table of power source descriptors. Each power source in the unit has an entry in this table.

3.4.2.1.3 unitPowerSourceEntry

An entry in the unit power source table.

3.4.2.1.4 psNumber

The power source number. Used as an index when accessing the unit power source table. Each power source in a unit has a unique number.

Note: Power source numbers should be allocated sequentially, starting from 1.

Syntax

IndexNumber [INTEGER (1..2147483647)]

A positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.2.1.5 psType

The power source type.

Syntax

PowerType [INTEGER { ac(1), dc(2), stored(3) }]

An enumeration identifying a power source type.

3.4.2.1.6 psStatus

The current status of this power source:

- `charged` indicates the power source is fully charged. For an external supply this is used to indicate that the supply voltage is above the threshold required by the unit.
- `charging` indicates that the power source is a battery (or other stored charge device) that is recharging itself from another power source but is not yet fully charged.
- `discharging` indicates that the power source is a battery (or other stored charge device) that is not recharging itself from another power source and is not fully charged or fully discharged.
- `discharged` indicates that the power source is fully discharged. For an external supply this is used to indicate that the supply voltage is below the threshold required by the unit.
- `faulty` indicates that a fault has been detected with the power source.
- `expired` indicates that the power source is a battery (or other stored charge device) which is due for replacement.

Syntax

PowerStatus [Integer { charged(1), charging(2), discharging(3), discharged(4), faulty(5), expired(6) }]

An enumeration identifying the status of a power source.

3.4.2.1.7 psChargeLevel

The current charge level of this power source as a percentage of full charge.

Syntax

ChargeLevel [INTEGER (0..100)]

A value representing the charge level of a battery or other stored charge device as a percentage.

3.4.2.1.8 psChargeTime

If the current status of this power source is `charging`, this indicates the estimated time (in minutes) until the power source is fully charged. If the current status of this power source is `discharging`, this indicates the estimated time (in minutes) until the power source is fully discharged. If the current status of this power source is any other value, this has the value `0`.

Syntax

CardinalNumber [INTEGER (0..2147483647)]

A zero or positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.3 Network/Transport Layers

3.4.3.1 Network

The group of objects in Table 3 shall be implemented by all compliant equipment that has a management model that incorporates one or more measurement information block(s). The root node for these objects shall be:

{ iso(1) standard(0) iec62379 measurement(7) measurementMIB(1) networkMeasurement(1) }

This node shall be used as the block type identifier for network measurement information blocks.

Table 3: Managed objects for network measurement information blocks

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
nMtBlockTable(1)	SEQUENCE OF NMtBlockEntry		none	none	no	m
└nMtBlockEntry(1)	NMtBlockEntry		none	none	no	m
├nMtBlockId(1)	BlockId	yes	none	none	no	m
├nMtIfIndex(2)	InterfaceIndex		listener	none	maybe	m
├nMtTxRxPoint(3)	TruthValue		listener	supervisor	no	m
├nMtNetworkType(4)	NetworkType		listener	none	maybe	m
├nMtTransportType(5)	TransportType		listener	none	maybe	m
├nMtTxRxAddr(6)	TAddress		listener	none	maybe	m
├nMtPortNumber(7)	CardinalNumber		listener	none	maybe	m
├nMtIGMPVersion(8)	CardinalNumber		listener	none	maybe	o
└nMtSIPServerAddr(9)	TAddress		listener	supervisor	maybe	o

3.4.3.1.1 nMtBlockTable

A table of network measurement block descriptors for this unit. Each network measurement block in the unit has a corresponding entry in this table.

3.4.3.1.2 nMtBlockEntry

An entry in the network measurement block table.

3.4.3.1.3 nMtBlockId

The block identifier for this block. Used as an index when accessing the network measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.3.1.4 nMtIfIndex

The identifier for the associated network interfaces object within MIB-II. This object shall be a copy of `ifIndex` (actually Textual Convention `InterfaceIndex` - see below from IF-MIB) from RFC 1213-MIB (MIB-II) (1.3.6.1.2.1.2.2.1.1) so as to provide a link between this network measurement block and the network interface it is associated with.

If `ifIndex` is not used within the unit, then an equivalent number to identify the interface should be used. This shall be greater than zero.

Syntax

InterfaceIndex [Integer32 (1..2147483647)]

(Note about and Description of Textual Convention `InterfaceIndex` from IF-MIB).

`InterfaceIndex` contains the semantics of `ifIndex` and should be used for any objects defined in other MIB modules that need these semantics.

A unique value, greater than zero, for each interface or interface sub-layer in the managed system. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

3.4.3.1.5 nMtTxRxPoint

This represents the measurement position in the media chain; whether the unit is a transmitter or receiver of the media. In the case of bi-directional media flow, then this value should be set to indicate the position from where the measurement is required to be made; from a media transmission or reception point of view.

- Transmitter (Tx) = `false`
- Receiver (Rx) = `true`

If the value of this object is `true` (Rx), then the receiver table (rxPointTable, § 3.4.4.3.1) shall exist, otherwise its existence is optional.

Syntax

TruthValue [INTEGER { true(1), false(2) }]

Represents a boolean value.

3.4.3.1.6 nMtNetworkType

A description of the current network type over which the media flow is flowing.

If the network type is `asi(3)`, then the remaining entries of

- nMtTransportType
- nMtTxRxAddr
- nMtPortNumber
- nMtIGMPVersion
- nMtSIPServerAddr

in this table are not required.

Syntax

NetworkType [INTEGER { ipv4(1), ipv6(2), asi(3) }]

An enumeration identifying the network type.

3.4.3.1.7 nMtTransportType

A description of the transport type being used to convey the media.

If `nMtNetworkType` is `asi(3)`, then `notApplicable(0)` shall be returned.

Syntax

TransportType [INTEGER { notApplicable(0), rtp(1) }]

An enumeration identifying the transport type.

Note: The values for this textual convention are NOT the same as the numbers used in the protocol field of IPv4 packets and the Next Header Field of IPv6 packets.

See http://www.iana.org/assignments/protocol-numbers

3.4.3.1.8 nMtTxRxAddr

The network address of either the send or receive point network interface over which the media is flowing. Which it is, is dependent on the measurement position in the media chain; whether the unit is a transmitter or receiver of the media. The value of `nMtTxRxPoint` shall determine which end is being referred to.

If `nMtNetworkType` is `asi(3)`, then `NULL` shall be returned.

Syntax

TAddress [OCTET STRING (SIZE (1..255))]

Denotes a transport service address. Imported from SNMPv2-TC.

3.4.3.1.9 nMtPortNumber

The port number assigned to the port over which the media stream or other application of interest is being measured.

A port number is a 16-bit unsigned integer, ranging from 0 to 65535.

If `nMtNetworkType` is `asi(3)`, then `zero` shall be returned.

Syntax

CardinalNumber [INTEGER (0..2147483647)]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.3.1.10 nMtIGMPVersion

The version of Internet Group Management Protocol (IGMP) being used.

If `nMtNetworkType` is `asi(3)`, then `zero` shall be returned.

If IGMP is not supported, then `zero` shall be returned.

Syntax

CardinalNumber [INTEGER (0..2147483647)]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.3.1.11 nMtSIPServerAddr

The IP address of the SIP server which the unit may be registered with.

If `nMtNetworkType` is `asi(3)`, then `NULL` shall be returned.

If the unit is not registered with a SIP server, then `NULL` shall be returned.

Syntax

TAddress [OCTET STRING (SIZE (1..255))]

Denotes a transport service address. Imported from SNMPv2-TC.

3.4.4 Application Layer

3.4.4.1 Audio

The group of objects in Table 4 shall be implemented by all compliant equipment that has a management model that incorporates one or more measurement information block(s). The root node for these objects shall be

{ iso(1) standard(0) iec62379 measurement(7) measurementMIB(1) audioMeasurement(2) }

This node shall be used as the block type identifier for audio measurement information blocks.

Table 4: Managed objects for audio measurement information blocks

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
aMtBlockTable(1)	SEQUENCE OF AMtBlockEntry		none	none	no	m
└aMtBlockEntry(1)	AMtBlockEntry		none	none	no	m
├aMtBlockId(1)	BlockId	yes	none	none	no	m
├aMtAudioComponentNumber(2)	IndexNumber	yes	none	none	maybe	m
├aMtNetworkBlockId(3)	BlockId		listener	none	no	m
├aMtAudioStatus(4)	TruthValue		listener	none	yes	m
├aMtAudioSignalFormat(5)	MediaFormat		listener	none	yes	m
├aMtAudioPId(6)	CardinalNumber		listener	none	yes	m
├aMtIfIndex(7)	InterfaceIndex		listener	none	maybe	m
├aMtFECType(8)	AudioFECType		listener	none	yes	o
└aMtFECLengthDimension(9)	IntegerNumber		listener	none	maybe	o

3.4.4.1.1 aMtBlockTable

A table of audio measurement block descriptors for this unit. Each audio measurement block in the unit has a corresponding entry in this table.

3.4.4.1.2 aMtBlockEntry

An entry in the audio measurement block table.

3.4.4.1.3 aMtBlockId

The block identifier for this block. Used as an index when accessing the audio measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.1.4 aMtAudioComponentNumber

The audio component identifier. The audio component number is the identifier for each audio item, when there are one or more audio items associated with a single video stream/signal.

For an audio only unit, this identifier shall also be used to identify each audio item, when there are one or more audio items present within the unit.

Note: An audio item here refers to a single audio stream/signal. Information pertaining to the number of channels within the audio item, such as stereo for two channels, for example, is contained within the 2.4.4.1.7 aMtAudioSignalFormat object.

Used as an index when accessing the audio measurement block table.

Syntax

IndexNumber [INTEGER (1..2147483647)]

A positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.4.1.5 aMtNetworkBlockId

The identifier for the associated network measurement block. This provides a link between the audio measurement block and the network measurement block.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.1.6 aMtAudioStatus

The status of the audio for this component number.

If `true`, indicates the audio signal is present. If `false`, indicates the audio signal is not present.

Syntax

TruthValue [INTEGER { true(1), false(2) }]

Represents a boolean value.

3.4.4.1.7 aMtAudioSignalFormat

A description of the current audio signal format.

If `aMtAudioPId > 0`, this should be the audio signal format associated with this PID.

If `aMtAudioPId = 0`, such as for an audio only unit that does not use PIDs, this object shall be the audio signal format of a single audio component.

If `aMtAudioStatus` is `true`, but no valid audio format can be identified, then the value `unspecifiedAudio` shall be returned.

If `aMtAudioStaus` is `false`, the value `noAudio` shall be returned.

Syntax

MediaFormat [OBJECT IDENTIFIER]

An object identifier identifying a defined media format. The format may be one defined in any Part of IEC 62379 or one defined elsewhere.

3.4.4.1.8 aMtAudioPId

The programme identifier for this particular audio component. For units that do not use PIDs, such as audio only units, the value of `zero` shall be returned.

Syntax

CardinalNumber [INTEGER (0..2147483647)]

A zero or positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.4.1.9 aMtIfIndex

The identifier for the associated network interfaces object within MIB-II. This object shall be a copy of `ifIndex` from RFC1213-MIB (MIB-II) (1.3.6.1.2.1.2.2.1.1) so as to provide a link between this audio component and the network interface on which it is present.

If `ifIndex` is not used within the unit, then an equivalent number to identify the interface should be used. This shall be > 0.

Syntax

InterfaceIndex [Integer32 (1..2147483647)]

(Description of InterfaceIndex from IF-MIB).

A unique value, greater than zero, for each interface or interface sub-layer in the managed system. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

3.4.4.1.10 aMtFECType

An indication as to the FEC type applied, if present.

Syntax

AudioFECType [INTEGER { none(0), smpte2021(1), smpte2022(2), rfc2733(3), proprietary(4) }]

An enumeration identifying the FEC type applied.

3.4.4.1.11 aMtFECLengthDimension

A description of the number of bytes over which FEC is applied and the matrix size being used. If the value of `aMtFECType` is `zero` (no FEC present), the value of `zero` shall be returned.

The format shall be `xxyy`, where, `xx` represents the number of bytes over which FEC is applied and `yy` represents the matrix size, for example, 2405 for 24 bytes and a 5x5 matrix.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.4.2 Video

The group of objects in Table 5 shall be implemented by all compliant equipment that has a management model that incorporates one or more measurement information block(s). The root node for these objects shall be:

{ iso(1) standard(0) iec62379 measurement(7) measurementMIB(1) videoMeasurement(3) }

This node shall be used as the block type identifier for video measurement information blocks.

Table 5: Managed objects for video measurement information blocks

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
vMtBlockTable(1)	SEQUENCE OF VMtBlockEntry		none	none	no	m
└vMtBlockEntry(1)	VMtBlockEntry		none	none	no	m
├-vMtBlockId(1)	BlockId	yes	none	none	no	m
├-vMtAudioBlockId(2)	BlockId		listener	none	maybe	m
├-vMtNetworkBlockId(3)	BlockId		listener	none	maybe	m
├-vMtVideoStatus(4)	TruthValue		listener	none	yes	m
├-vMtVideoSourceFormat(5)	MediaFormat		listener	none	yes	m
├-vMtVideoCodingType(6)	MediaFormat		listener	none	yes	m
├-vMtVideoBitRateType(7)	BitRateType		listener	none	yes	m
├-vMtVideoBitRate(8)	CardinalNumber		listener	none	yes	m
├-vMtAspectRatio(9)	MediaFormat		listener	none	yes	m
├-vMtFECType(10)	VideoFECType		listener	none	yes	o
├-vMtFECLengthDimension(11)	IntegerNumber		listener	none	maybe	o
└-vMtTrickModeSupport(12)	TruthValue		listener	none	yes	o

3.4.4.2.1 vMtBlockTable

A table of video measurement block descriptors for this unit. Each video measurement block in the unit has a corresponding entry in this table.

3.4.4.2.2 vMtBlockEntry

An entry in the video measurement block table.

3.4.4.2.3 vMtBlockId

The block identifier for this block. Used as an index when accessing the video measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.2.4 vMtAudioBlockId

The identifier for the associated audio measurement block. This object provides a link between the video measurement block and the audio measurement block.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.2.5 vMtNetworkBlockId

The identifier for the associated network measurement block. This object provides a link between the video measurement block and the network measurement block.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.2.6 vMtVideoStatus

The status of the video signal.

If `true`, indicates the video signal is present. If `false`, indicates the video signal is not present.

Syntax

TruthValue [INTEGER { true(1), false(2) }]

Represents a boolean value.

3.4.4.2.7 vMtVideoSourceFormat

A description of the structure of the base video source format.

The format is defined in IEC62397-3 Ed.1

It has the following format:

1.0.62379.3.2.1.3.w.x.y.z where

- w is the frame rate in Hz
- x is the definition of the video source type
    - Unspecified (0)
    - SD (1)
    - HD (2)
- y is the number of vertical lines of resolution
- z is the video scan type
    - Unspecified (0)
    - Progressive - P (1)
    - Interlaced - I (2)
    - Progressive Segmented Frame – PSF (3)

Syntax

MediaFormat [OBJECT IDENTIFIER]

An object identifier identifying a defined media format. The format may be one defined in any Part of IEC 62379 or one defined elsewhere.

3.4.4.2.8 vMtVideoCodingType

A description of the current video signal coding type.

The format is defined in IEC62397-3 Ed.1

It has the following format:

1.0.62379.3.2.1.4.z where

z is either uncompressed or the coding type

- Unspecified (0)
- Uncompressed (1)
- MPEG2 (2)
- H264 (3)
- JPEG2000 (4)
- SMPTE VC-2 (5)
- VP8 (6)
- H264 Scalable Extension (7)

Syntax

MediaFormat [OBJECT IDENTIFIER]

An object identifier identifying a defined media format. The format may be one defined in any Part of IEC 62379 or one defined elsewhere.

3.4.4.2.9 vMtVideoBitRateType

A description of the current video bit rate type applied.

Syntax

BitRateType [INTEGER { unspecified(0), vbr(1), cbr(2) }]

An enumeration identifying the video bit rate type applied.

- vbr = variable bit rate
- cbr = constant bit rate

3.4.4.2.10 vMtVideoBitRate

A description of the current video bit rate in kbit/s.

- If `vMtVideoBitRateType = Unspecified (0)`, then this object shall be set to a default value of `Unspecified (0)`
- If `vMtVideoBitRateType = VBR (1)`, then this object shall be the advertised Maximum Bit Rate
- If `vMtVideoBitRateType = CBR (2)`, then this object shall be the advertised (Constant) Bit Rate

Syntax

CardinalNumber [INTEGER (0..2147483647)]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.4.2.11 vMtVideoAspectRatio

A description of the current video aspect ratio.

The format is defined in IEC62397-3 Ed.1

It has the following format:

1.0.62379.3.2.1.5.y.z where

- y is the source aspect ratio
  - Unspecified (0)
  - 4:3 (43)
  - 16:9 (169)
  - 2.21:1 (221)

  Note: Unspecified (0) uses only the undefined AFD code 0000.
- z is the active format description code for the source aspect ratio
  - The codes are from 0000-1111
  - See SMPTE ST 2016-1:2009 for code descriptions.

Syntax

MediaFormat [OBJECT IDENTIFIER]

An object identifier identifying a defined media format. The format may be one defined in any Part of IEC 62379 or one defined elsewhere.

3.4.4.2.12 vMtFECType

An indication as to the FEC type applied, if present.

Syntax

VideoFECType [INTEGER { none(0), smpte2022(2), proprietary(4) }]

An enumeration identifying the FEC type applied.

3.4.4.2.13 vMtFECLengthDimension

A description of the number of bytes over which FEC is applied and the matrix size being used. If the value of `vMtFECType` is `zero` (no FEC present), the value of `zero` shall be returned.

The format shall be `xxyy`, where, `xx` represents the number of bytes over which FEC is applied and `yy` represents the matrix size, for example, 2405 for 24 bytes and a 5x5 matrix.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.4.2.14 vMtTrickModeSupport

An indication as to whether trick mode is supported or not.

If `true`, indicates trick mode is supported. If `false`, indicates trick mode is not supported.

Syntax

TruthValue [INTEGER { true(1), false(2) }]

Represents a boolean value.

3.4.4.3 Receiver

The group of objects in Table 6 shall be implemented by all compliant equipment that has a management model that incorporates one or more measurement information block(s). The root node for these objects shall be:

{ iso(1) standard(0) iec62379 measurement(7) measurementMIB(1) receiverMeasurement(4) }

Note: This group of objects shall only exist if the value of `nMtTxRxPoint` is `true` in the corresponding row in the `nMtTable`.

This node shall be used as the block type identifier for receiver measurement information blocks.

Table 6: Managed objects for receiver measurement information blocks

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
rxPointTable(2)	SEQUENCE OF RxPointEntry		none	none	maybe	m
└ rxPointEntry(1)	RxPointEntry		none	none	maybe	m
├ rxPointBlockId(1)	BlockId	yes	none	none	maybe	m
├ rxPointNetworkBlockId(2)	BlockId	yes	none	none	maybe	m
├ rxPointBufferSize(3)	BufferSize		listener	operator	maybe	m
├ rxPointBufferOcpancyTime(4)	BufferOcpncyTime		listener	none	maybe	m
├ rxPointBufferOcpncyPcnt(5)	BufferOcpncyPercent		listener	none	maybe	m
├ rxPointMDI(6)	Utf8String		listener	none	maybe	o
└ rxPointTSDF(7)	CardinalNumber		listener	none	maybe	o

3.4.4.3.1 rxPointTable

A table of receiver point measurement block descriptors for this unit. Each receiver point measurement block in the unit has a corresponding entry in this table.

Entries in this table shall only exist if the value of `nMtTxRxPoint` is `true` in the corresponding row in the `nMtTable`.

The number of entries is between `zero` and the number of entries in the `nMtTable`. Since an entry in the `nMtTable` may not have the value of true for `nMtTxRxPoint`, there may not be an entry in this table for each entry in the `nMtTable`.

In addition to the index of `rxPointBlockId`, this table also uses the same index of `nMtBlockId` as the `nMtTable`.

3.4.4.3.2 rxPointEntry

An entry in the receiver point measurement block table.

3.4.4.3.3 rxPointBlockId

The block identifier for this block. Used as an index when accessing the receiver point measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.3.4 rxPointNetworkBlockId

The index value (`nMtBlockId`) from the Network Measurement table (`nMtTable`) used as an index when accessing the `rxPoint` measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.3.5 rxPointBufferSize

A description of the current total size of the receive buffer in ms.

Syntax

BufferSize [Unsigned32]

3.4.4.3.6 rxPointBufferOcpancyTime

A description of the amount of data, in ms, occupying the receive buffer. This ranges between `zero` and `rxPointBufferSize`.

Syntax

BufferOcpncyTime [Gauge32]

The use of this type will allow changes (either up or down) to be reported, but limited to between `zero` and the value of `rxPointBufferSize`.

3.4.4.3.7 rxPointBufferOcpncyPcnt

A description of the amount of data, expressed as a percentage of the total receive buffer size, occupying the receive buffer.

Syntax

BufferOcpncyPercent [INTEGER (0..100)]

3.4.4.3.8 rxPointMDI

A description of the measured and calculated Media Delivery Index (MDI). The MDI has two components, the Delay Factor (DF) and the Media Loss Rate (MLR).

MDI is expressed as simple pairs of absolute values in the form of DF:MLR, e.g. 06:10

The measurement units of DF and MLR are ms.

Syntax

Utf8String [OCTET STRING (SIZE (0..80))]

A UTF-8 encoded text string.

3.4.4.3.9 rxPointTSDF

A description of the Time-stamped Delay Factor (TS-DF).

Time-stamped Delay Factor is calculated as:

TS-DF = D(Max) - D(Min)

The measurement units of TS and DF are ms.

For further details of this, see EBU Tech 3337.

Syntax

CardinalNumber [INTEGER (0..2147483647)]

A zero or positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.4.4 Temperature

The group of objects in Table 7 shall be implemented by all compliant equipment that has a management model that incorporates one or more measurement information block(s). The root node for these objects shall be:

{ iso(1) standard(0) iec62379 measurement(7) measurementMIB(1) temperatureMeasurement(5) }

Note: This group of objects are optional, but mandatory if temperature sensors are present within the unit.

Table 7: Managed objects conveying temperature information about the unit

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
temperatureTable(1)	SEQUENCE OF TemperatureEntry		none	none	no	m
└ temperatureEntry(1)	TemperatureEntry		none	none	no	m
├ temperatureBlockId(1)	BlockId	yes	none	none	no	m
├ temperatureLocnNumber(2)	IndexNumber	yes	none	none	no	m
├ temperatureLocn(3)	TemperatureLocn		listener	none	no	m
├ temperatureTrend(4)	TemperatureTrend		listener	none	yes	m
├ temperatureStatus(5)	TemperatureStatus		listener	none	yes	m
├ temperatureLowWarning(6)	IntegerNumber		listener	none	yes	m
├ temperatureHighWarning(7)	IntegerNumber		listener	none	yes	m
├ temperatureLowCritical(8)	IntegerNumber		listener	none	yes	m
└ temperatureHighCritical(9)	IntegerNumber		listener	none	yes	m

3.4.4.4.1 temperatureTable

A table of temperature measurement descriptors. Each temperature measurement point in the unit has an entry in this table.

3.4.4.4.2 temperatureEntry

An entry in the temperature measurement table.

3.4.4.4.3 temperatureBlockId

The block identifier for this block. Used as an index when accessing the temperature measurement block table.

Syntax

BlockId [INTEGER (1..2147483647)]

A handle uniquely identifying a control block within a unit.

3.4.4.4.4 temperatureLocnNumber

The identifier where the temperature is measured. Used as an index when accessing the unit temperature measurement table. Each temperature measurement point in a unit has a unique number.

Note: temperatureLocnNumber should be allocated sequentially, starting from 1.

Syntax

IndexNumber [INTEGER (1..2147483647)]

A positive integer value with no specific upper limit. The upper limit specified here is purely for SMIv2 compatibility.

3.4.4.4.5 temperatureLocation

A text description of the location of where the temperature is being measured.

Syntax

TemperatureLocn [OCTET STRING (SIZE (0..80))]

A UTF-8 encoded text string.

3.4.4.4.6 temperatureTrend

The current temperature measured at this location. Unit is Degrees Celsius.

Syntax

TemperatureTrend [Gauge32]

The use of this type will allow changes (either up or down) to be reported.

3.4.4.4.7 temperatureStatus

The current status of this particular temperature measurement.

- `undetermined` indicates the temperature cannot be determined
- `other` indicates the temperature is in some other state
- `unknown` indicates the temperature is unknown
- `ok` indicates the temperature is within acceptable limits
- `warning` indicates the temperature has reached the warning level for the unit
- `critical` indicates the temperature has reached the critical level for the unit
- `non-recoverable` indicates the temperature level for the unit is no longer recoverable

Syntax

temperatureStatus [INTEGER { undetermined(0), other(1), unknown(2), ok(3), warning(4), critical(5), non-recoverable(6) }]

An enumeration identifying the status of the temperature sensor.

3.4.4.4.8 temperatureLowWarning

The low warning limit for the measured temperature. Unit is Degrees Celsius.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for

SMIv2 compatibility.

3.4.4.4.9 temperatureHighWarning

The high warning limit for the measured temperature. Unit is Degrees Celsius.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.4.4.10 temperatureLowCritical

The low critical limit for the measured temperature. Unit is Degrees Celsius.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

3.4.4.4.11 temperatureHighCritical

The high critical limit for the measured temperature. Unit is Degrees Celsius.

Syntax

IntegerNumber [Integer32]

An integer value with no specific lower or upper limit. The limits specified here are purely for SMIv2 compatibility.

4. References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

1. IEC 62379-1:2007 Common Control Interface for Networked Audio and Video Products – Part 1: General

2. IEC 62379-2:2008 Common Control Interface for Networked Audio and Video Products – Part 2: Audio

3. IEC 62379-3 Ed.1: Common Control Interface for Networked Audio and Video Products – Part 3: Video[5]

4. IEC 62379-7: Common Control Interface for Networked Audio and Video Products – Part 7: Measurement[6]

[5] This document is currently in draft form. As of the date of this document (EBU Tech 3345), IEC 62379-3 Ed.1 contains only the video formats as shown in Annex F.

[6] This document is currently in draft form. As of the date of this document (EBU Tech 3345), IEC 62379-7 is expected to go forward for Final Draft International Standard application within Q3 of 2011.

Annex A: (Informative) MIB Definitions from IEC 62379-1

This annex provides reference information on the attributes within each of the tables, extracted from IEC 62379-1, § 4.2.

All network-managed control functions in conformant equipment shall be represented as instances of managed object types defined in this standard or other parts of this standard or elsewhere.

Note 1: The use of managed object types not defined by IEC 62379 is permitted to allow control of functions outside the scope of IEC 62379 using the standard protocol. Object types defined by IEC 62379 should be used where applicable, in preference to object types defined elsewhere.

Note 2: Managed objects are typically organized in groups of related functions.

Each managed object type in this or other parts of IEC 62379 is defined by the following attributes.

- *Identifier* specifies the name and number that identify the object relative to the group or object from which it descends.
- *Syntax* specifies the syntax of the abstract data structure representing the object value. The syntax is described using abstract syntax notation 1 (ASN.1), as specified in ISO/IEC 8824-1:1998.
- *Index* specifies whether the object is used to uniquely identify a row in a managed object table.
- *Readable* specifies the privilege level (as defined below) for read access to the object. A management call with a privilege level greater than, or equal to, this level shall be permitted to perform a get or get-next operation on the object. A read access level of none specifies that get and get-next operations are not permitted on the object.
- *Writable* specifies the privilege level (as defined below) for write access to the object. A management call with a privilege level greater than, or equal to, this level shall be permitted to perform a set operation on the object. A write access level of none specifies that set operations are not permitted on the object.
- *Volatile* specifies whether the current value of the object is retained after a hard reset or period of power loss:
    - *no* indicates that the value of the object is either built in to the unit or stored in non-volatile memory;
    - *yes* indicates that the value of the object is transient;
    - maybe indicates that the value of the object may or may not be volatile, depending on the design of the unit or on the value of some other object in the MIB.
- *Status* specifies the required level of implementation support for the object:
    - *mandatory* indicates that the object shall be implemented by all conformant equipment that also implements that object's parent group or object;
    - *optional* indicates that the object may or may not be implemented by any conformant equipment that also implements that object's parent group or object.
    - *deprecated* indicates that the object shall not be implemented by any newly designed conformant equipment.

For brevity, the status attributes are abbreviated to *m*, *o*, and *d* in object definition tables.

The privilege levels used for the *readable* and *writable* attributes are, from lowest to highest:

- *listener*
- *operator*
- *supervisor*
- *maintenance*

From IEC62379-1, an extract from section, **0.7 Privilege levels**:

Listener is the lowest privilege level, intended for use by those who wish to monitor audio or video signals passing through the unit (for example, someone who wishes to listen in on the output of a studio via their PC). Listeners can set up calls from audio and video sources to equipment which is local to them but cannot change anything that would affect the experience of other users.

Operator is the next privilege level, intended for use by those who are controlling the day-to- day operation of the unit (for example, a studio technical operator). Operators can change things which affect other users, such as the mix of signals that provides the output from a studio, or by issuing "pause", "seek", etc. commands to play-out equipment.

Supervisor is the next privilege level, intended for use by those who are controlling and maintaining the network such as a control room technical operator.

Maintenance is the highest privilege level, intended for use by those who need to perform tasks that might disrupt normal operation of the unit, such as updating firmware or causing the unit to enter a diagnostic mode.

Annex B: (Informative) Type and Sequence definitions from IEC 62379-1

This annex provides a reference to any type definitions and sequences defined in IEC 62379-1 that are used within this standard. If there is any inconsistency between this annex and those in the appropriate section of IEC 62379-1, then IEC 62379-1 takes precedence.

The format used to describe the MIB objects conforms to IETF STD 58 (SMIv2).

```
PowerType::= INTEGER {
    ac     (1), -- external AC power supply
    dc     (2), -- external DC power supply
    stored (3)   -- internal battery or other charge storage device
  } (ac..stored)
  -- An enumeration identifying a power source type.

  PowerStatus::= INTEGER {
    charged     (1),
    charging    (2),
    discharging (3),
    discharged  (4),
    faulty      (5),
    expired     (6)
  } (charged..expired)
  -- An enumeration identifying the status of a power source.

  ChargeLevel::= INTEGER (0..100)
  -- A value representing the charge level of a battery or other stored
  -- charge device as a percentage.

UnitPowerSourceEntry::= SEQUENCE {
    psNumber       IndexNumber,
    psType         PowerType,
    psStatus       PowerStatus,
    psChargeLevel  ChargeLevel,
    psChargeTime   CardinalNumber
  }

  BlockEntry::= SEQUENCE {
    blockId    BlockId,
    blockType  BlockType
  }
```

Annex C: (Informative) Machine-readable measurement block definitions

This annex provides a machine-readable version of the measurement block definitions which is intended to be interpretable by standard MIB browsing software tools. It does not express all the requirements of the Standard, for instance where access to an object is restricted at certain privilege levels. If there is any inconsistency between this annex and the appropriate clauses in IEC 62379-7, the appropriate clauses in IEC 62379-7 take precedence.

The format used to describe the MIB objects conforms to IETF STD 58 (SMIv2).

```
IEC62379-7-IPM-MIB DEFINITIONS ::= BEGIN

   IMPORTS
      iec62379, unitPowerSource, psNumber, psType, psStatus,
      psChargeLevel, psChargeTime, blockId, blockType, BlockId,
      BlockType, CardinalNumber, ClockSource, IndexNumber, IntegerNumber,
      MediaFormat, Utf8String
         FROM IEC62379-1-MIB
      NetworkType, TransportType, TemperatureStatus, TemperatureLocn, TemperatureTrend,
      AudioFECType, VideoFECType, BufferSize, BufferOcpncyPercent, BufferOcpncyTime,
      BitRateType
         FROM IEC62379-7-IPM-TC-MIB
      InterfaceIndex
         FROM IF-MIB
      OBJECT-GROUP, MODULE-COMPLIANCE
         FROM SNMPv2-CONF
      OBJECT-TYPE, MODULE-IDENTITY
         FROM SNMPv2-SMI
      TAddress, TruthValue
         FROM SNMPv2-TC;

   -- 1.0.62379.7.1
   measurementMIB MODULE-IDENTITY
      LAST-UPDATED "201106211200Z"    -- June 21, 2011 at 12:00 GMT
      ORGANIZATION
         "IEC PT62379"
      CONTACT-INFO
         "Not specified."
      DESCRIPTION
         "The MIB module for managing measurement functions
         in IEC 62379 compliant equipment.
         The contents of this MIB Module have been defined by
         the EBU ECN-IPM (European Broadcasting Union
         Expert Communities Networks, IP Measurements) group."
      REVISION "201106211200Z"     -- June 21, 2011 at 12:00 GMT
      DESCRIPTION
         "Added additional coding types VP8 and H264 Scalable Extension
         entries.
         Added video bit rate type and rate.
         Addition of Aspect Ratio to the Video Measurement Table.
         Move FEC information down to accommodate Aspect Ratio."
      ::= { measurement 1 }
```

```
--
-- Node definitions
--

     -- 1.0.62379.7
     measurement OBJECT IDENTIFIER ::= { iec62379 7 }


     -- 1.0.62379.7.1.0
     measurementMIBCompliance OBJECT IDENTIFIER ::= { measurementMIB 0 }


     -- 1.0.62379.7.1.0.1
     measurementMIBComplianceV1 MODULE-COMPLIANCE
        STATUS current
        DESCRIPTION
           " The compliance statement for entities that conform to
           IEC 62379-7 (2010)."
        MODULE -- this module
           MANDATORY-GROUPS { audioMeasurementGroup, networkMeasurementGroup,
receiverMeasurementGroup }
           GROUP videoMeasurementGroup
              DESCRIPTION
                 "Mandatory for equipment that has video functionality."
           GROUP temperatureMeasurementGroup
              DESCRIPTION
                 "Mandatory for equipment that has temperature measurement functionality."
        ::= { measurementMIBCompliance 1 }


     -- 1.0.62379.7.1.1
     networkMeasurement OBJECT IDENTIFIER ::= { measurementMIB 1 }


     -- 1.0.62379.7.1.1.1
     nMtTable OBJECT-TYPE
        SYNTAX SEQUENCE OF NMtEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
           "Description."
        ::= { networkMeasurement 1 }


     -- 1.0.62379.7.1.1.1.1
     nMtEntry OBJECT-TYPE
        SYNTAX NMtEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
           "Description."
        INDEX { nMtBlockId }
        ::= { nMtTable 1 }


     NMtEntry ::=
```

```
   SEQUENCE {
      nMtBlockId
         BlockId,
      nMtIfIndex
         InterfaceIndex,
      nMtTxRxPoint
         TruthValue,
      nMtNetworkType
         NetworkType,
      nMtTransportType
         TransportType,
      nMtTxRxAddr
         TAddress,
      nMtPortNumber
         CardinalNumber,
      nMtIGMPVersion
         CardinalNumber,
      nMtSIPServerAddr
         TAddress
    }

-- 1.0.62379.7.1.1.1.1.1
nMtBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The block identifier for this block.
      Used as an index when accessing the network measurement
      block table."
   ::= { nMtEntry 1 }


-- 1.0.62379.7.1.1.1.1.2
nMtIfIndex OBJECT-TYPE
   SYNTAX InterfaceIndex
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The identifier for the associated network interfaces object
      within MIB-II.

      This object shall be a copy of ifIndex (actually Textual
      Convention InterfaceIndex - see below from IF-MIB) from
      RFC1213-MIB (MIB-II) (1.3.6.1.2.1.2.2.1.1) so as to provide
      a link between this network measurement block and the
      network interface it is associated with.

      If ifIndex is not used within the unit, then an equivalent
      number to identify the interface should be used.
      This shall be greater than zero.

      (Description of ifIndex from RFC1213-MIB).
      A unique value for each interface.  Its value
      ranges between 1 and the value of ifNumber (1.3.6.1.2.1.2.1).
      The value for each interface must remain constant at
      least from one re-initialization of the entity's
```

```
      network management system to the next re-
      initialization.

      (Note about and Description of Textual Convention
      InterfaceIndex from IF-MIB).

      InterfaceIndex contains the semantics of ifIndex and
      should be used for any objects defined in other MIB modules
      that need these semantics.

      A unique value, greater than zero, for each interface or
      interface sub-layer in the managed system.  It is
      recommended that values are assigned contiguously starting
      from 1.  The value for each interface sub-layer must remain
      constant at least from one re-initialization of the entity's
      network management system to the next re-initialization."
   ::= { nMtEntry 2 }


-- 1.0.62379.7.1.1.1.1.3
nMtTxRxPoint OBJECT-TYPE
   SYNTAX TruthValue
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "This represents the measurement position in the media chain;
      whether the unit is a transmitter or receiver of the
      media.

      In the case of bi-directional media flow, then this value
      should be set to indicate the position from where the
      measurement is required to be made; from a media
      transmission or reception point of view.

      Transmitter (Tx) = false
      Receiver (Rx) = true

      If the value of this object is true (Rx), then the receiver table
      (rxPointTable) shall exist, otherwise its existence is optional."
   ::= { nMtEntry 3 }


-- 1.0.62379.7.1.1.1.1.4
nMtNetworkType OBJECT-TYPE
   SYNTAX NetworkType
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current network type over which the
      media flow is flowing.

      Current Values are:
      ipv4(1),
      ipv6(2),
      asi(3)

      If the network type is asi(3), then the remaining entries
```

```
      of

      nMtTransportType
      nMtTxRxAddr
      nMtPortNumber
      nMtIGMPVersion
      nMtSIPServerAddr

      in this table are not required."
   ::= { nMtEntry 4 }


-- 1.0.62379.7.1.1.1.1.5
nMtTransportType OBJECT-TYPE
   SYNTAX TransportType
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the transport type being used to convey the
      media.

      If nMtNetworkType is asi (3), then notApplicable(0) shall be
      returned.

      Current Values are:

      notApplicable(0),
      rtp(1)"
   ::= { nMtEntry 5 }


-- 1.0.62379.7.1.1.1.1.6
nMtTxRxAddr OBJECT-TYPE
   SYNTAX TAddress
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The network address of either the send or receive point
      network interface over which the media is flowing.
      Which it is, is dependent on the measurement position in
      the media chain; whether the unit is a transmitter or
      receiver of the media.
      The value of nMtTxRxPoint shall determine which end is
      being referred to.

      If nMtNetworkType is asi (3), then NULL shall be returned."
   ::= { nMtEntry 6 }


-- 1.0.62379.7.1.1.1.1.7
nMtPortNumber OBJECT-TYPE
   SYNTAX CardinalNumber
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The port number assigned to the port over which the media
      stream or other application of interest is being measured.
```

```
      A port number is a 16-bit unsigned integer, ranging from
      0 to 65535.

      If nMtNetworkType is asi (3), then zero shall be returned."
   ::= { nMtEntry 7 }


-- 1.0.62379.7.1.1.1.1.8
nMtIGMPVersion OBJECT-TYPE
   SYNTAX CardinalNumber
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The version of Internet Group Management Protocol (IGMP)
      being used.

      If nMtNetworkType is asi (3), then zero shall be returned.

      If IGMP is not supported, then zero shall be returned."
   ::= { nMtEntry 8 }


-- 1.0.62379.7.1.1.1.1.9
nMtSIPServerAddr OBJECT-TYPE
   SYNTAX TAddress
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "The IP address of the SIP server which the unit may be
      registered with.

      If nMtNetworkType is asi (3), then NULL shall be returned.

      If the unit is not registered with a SIP server, then NULL
      shall be returned."
   ::= { nMtEntry 9 }


-- 1.0.62379.7.1.1.2
networkMeasurementGroup OBJECT-GROUP
   OBJECTS { nMtIfIndex, nMtTxRxPoint, nMtNetworkType, nMtTransportType, nMtTxRxAddr,
      nMtPortNumber, nMtIGMPVersion, nMtSIPServerAddr }
   STATUS current
   DESCRIPTION
      "The group of objects used for network measurements."
   ::= { networkMeasurement 2 }


-- 1.0.62379.7.1.2
audioMeasurement OBJECT IDENTIFIER ::= { measurementMIB 2 }


-- 1.0.62379.7.1.2.1
aMtBlockTable OBJECT-TYPE
   SYNTAX SEQUENCE OF AMtBlockEntry
   MAX-ACCESS not-accessible
```

```
   STATUS current
   DESCRIPTION
      "Description."
   ::= { audioMeasurement 1 }


-- 1.0.62379.7.1.2.1.1
aMtBlockEntry OBJECT-TYPE
   SYNTAX AMtBlockEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "Description."
   INDEX { aMtBlockId, aMtAudioComponentNumber }
   ::= { aMtBlockTable 1 }


AMtBlockEntry ::=
   SEQUENCE {
      aMtBlockId
         BlockId,
      aMtAudioComponentNumber
         IndexNumber,
      aMtNetworkBlockId
         BlockId,
      aMtAudioStatus
         TruthValue,
      aMtAudioSignalFormat
         MediaFormat,
      aMtAudioPId
         CardinalNumber,
      aMtIfIndex
         InterfaceIndex,
      aMtFECType
         AudioFECType,
      aMtFECLengthDimension
         IntegerNumber
    }

-- 1.0.62379.7.1.2.1.1.1
aMtBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The block identifier for this block.
      Used as an index when accessing the audio measurement
      block table."
   ::= { aMtBlockEntry 1 }


-- 1.0.62379.7.1.2.1.1.2
aMtAudioComponentNumber OBJECT-TYPE
   SYNTAX IndexNumber
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
```

```
      "The audio component identifier.
      The audio component number is the identifier for each audio
      item, when there are one or more audio items associated
      with a single video stream/signal.

      For an audio only unit this identifier shall also be
      used to identify each audio item, when there are one or more
      audio items present within the unit.

      Used as an index when accessing the audio measurement
      block table."
   ::= { aMtBlockEntry 2 }


-- 1.0.62379.7.1.2.1.1.3
aMtNetworkBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The identifier for the associated network measurement block.
      This provides a link between the audio measurement block
      and the network measurement block."
   ::= { aMtBlockEntry 3 }


-- 1.0.62379.7.1.2.1.1.4
aMtAudioStatus OBJECT-TYPE
   SYNTAX TruthValue
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The status of the audio for this component number.

      If true, indicates the audio signal is present.
      If false, indicates the audio signal is not present."
   ::= { aMtBlockEntry 4 }


-- 1.0.62379.7.1.2.1.1.5
aMtAudioSignalFormat OBJECT-TYPE
   SYNTAX MediaFormat
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current audio signal format.

      If aMtAudioPId > 0, this should be the audio signal format
      associated with this PID.

      If aMTAudioPID = 0, such as for an audio only device that does
      not use PIDs, this object shall be the audio signal format of a
      single audio component.

      If aMtAudioStatus is true, but no valid audio format can be
      identified, then the value unspecifiedAudio shall be returned.
```

```
      If aMtAudioStaus is false, the value noAudio shall be returned."
   ::= { aMtBlockEntry 5 }


-- 1.0.62379.7.1.2.1.1.6
aMtAudioPId OBJECT-TYPE
   SYNTAX CardinalNumber
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The programme identifier for this particular audio component.

      For units that do not use PIDs, such as audio only units,
      the value of zero shall be returned."
   ::= { aMtBlockEntry 6 }


-- 1.0.62379.7.1.2.1.1.7
aMtIfIndex OBJECT-TYPE
   SYNTAX InterfaceIndex
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The identifier for the associated network interfaces object
      within MIB-II.

      This object shall be a copy of ifIndex from RFC1213-MIB (MIB-II)
      (1.3.6.1.2.1.2.2.1.1) so as to provide a link between this
      audio component and the network interface on which
      it is present.

      If ifIndex is not used within the unit, then an equivalent
      number to identify the interface should be used.
      This shall be > 0.

      (Description of ifIndex from RFC1213-MIB).
      A unique value for each interface.  Its value
      ranges between 1 and the value of ifNumber (1.3.6.1.2.1.2.1).
      The value for each interface must remain constant at
      least from one re-initialization of the entity's
      network management system to the next re-
      initialization.

      (Description of InterfaceIndex from IF-MIB).
      A unique value, greater than zero, for each interface or
      interface sub-layer in the managed system.  It is
      recommended that values are assigned contiguously starting
      from 1.  The value for each interface sub-layer must remain
      constant at least from one re-initialization of the entity's
      network management system to the next re-initialization."
   ::= { aMtBlockEntry 7 }


-- 1.0.62379.7.1.2.1.1.8
aMtFECType OBJECT-TYPE
   SYNTAX AudioFECType
   MAX-ACCESS read-only
```

```
        STATUS current
        DESCRIPTION
          "An indication as to the FEC type applied, if present.

          Current Values are:

          none(0),
          smpte2021(1),
          smpte2022(2),
          rfc2733(3),
          proprietary(4)"
        ::= { aMtBlockEntry 8 }


      -- 1.0.62379.7.1.2.1.1.9
      aMtFECLengthDimension OBJECT-TYPE
        SYNTAX IntegerNumber
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
          "A description of the number of bytes over which FEC is
          applied and the matrix size being used.

          If the value of aMtFECType is zero (no FEC present), the
          value of zero shall be returned.

          The format shall be xxyy, where, xx represents the number
          of bytes over which FEC is applied and yy represents the
          matrix size, for example, 2405 for 24 bytes and a 5x5 matrix.
          "
        ::= { aMtBlockEntry 9 }


      -- 1.0.62379.7.1.2.2
      audioMeasurementGroup OBJECT-GROUP
        OBJECTS { aMtNetworkBlockId, aMtAudioStatus, aMtAudioSignalFormat, aMtAudioPId,
aMtIfIndex,
          aMtFECType, aMtFECLengthDimension }
        STATUS current
        DESCRIPTION
          "The group of objects used for audio measurements."
        ::= { audioMeasurement 2 }


      -- 1.0.62379.7.1.3
      videoMeasurement OBJECT IDENTIFIER ::= { measurementMIB 3 }


      -- 1.0.62379.7.1.3.1
      vMtTable OBJECT-TYPE
        SYNTAX SEQUENCE OF VMtEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
          "Description."
        ::= { videoMeasurement 1 }
```

```
-- 1.0.62379.7.1.3.1.1
vMtEntry OBJECT-TYPE
   SYNTAX VMtEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "Description."
   INDEX { vMtBlockId }
   ::= { vMtTable 1 }


VMtEntry ::=
   SEQUENCE {
      vMtBlockId
         BlockId,
      vMtAudioBlockId
         BlockId,
      vMtNetworkBlockId
         BlockId,
      vMtVideoStatus
         TruthValue,
      vMtVideoSourceFormat
         MediaFormat,
      vMtVideoCodingType
         MediaFormat,
      vMtVideoBitRateType
         BitRateType,
      vMtVideoBitRate
         CardinalNumber,
      vMtAspectRatio
         MediaFormat,
      vMtFECType
         VideoFECType,
      vMtFECLengthDimension
         IntegerNumber,
      vMtTrickModeSupport
         TruthValue
    }

-- 1.0.62379.7.1.3.1.1.1
vMtBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The block identifier for this block.
      Used as an index when accessing the video measurement
      block table."
   ::= { vMtEntry 1 }


-- 1.0.62379.7.1.3.1.1.2
vMtAudioBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS read-only
   STATUS current
```

```
   DESCRIPTION
      "The identifier for the associated audio measurement block.
      This object provides a link between the video measurement
      block and the audio measurement block."
   ::= { vMtEntry 2 }


-- 1.0.62379.7.1.3.1.1.3
vMtNetworkBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The identifier for the associated network measurement block.
      This object provides a link between the video measurement
      block and the network measurement block."
   ::= { vMtEntry 3 }


-- 1.0.62379.7.1.3.1.1.4
vMtVideoStatus OBJECT-TYPE
   SYNTAX TruthValue
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The status of the video signal.

      If true, indicates the video signal is present.
      If false, indicates the video signal is not present."
   ::= { vMtEntry 4 }


-- 1.0.62379.7.1.3.1.1.5
vMtVideoSourceFormat OBJECT-TYPE
   SYNTAX MediaFormat
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the structure of the base video source format.

      The format is defined in IEC62397-3 Ed.1
      It has the following format:

      1.0.62379.3.2.1.3.w.x.y.z where

      w is the frame rate in Hz

      x is the definition of the video source type
      Unspecified (0),
      Standard Definition - SD (1),
      High Definition - HD (2)

      y is the number of vertical lines of resolution

      z is the video scan type
      Unspecified (0),
      Progressive (1),
```

```
      Interlaced (2),
      Progressive Segmented Frame - PSF (3)"
   ::= { vMtEntry 5 }


-- 1.0.62379.7.1.3.1.1.6
vMtVideoCodingType OBJECT-TYPE
   SYNTAX MediaFormat
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current video signal coding type.

      The format is defined in IEC62397-3 Ed.1
      It has the following format:

      1.0.62379.3.2.1.4.z where

      z is either uncompressed or the coding type

      Unspecified (0),
      Uncompressed (1),
      MPEG2 (2),
      H264 (3),
      JPEG2000 (4),
      SMPTE VC-2 (5),
      VP8 (6),
      H264ScaleExtn (7)"
   ::= { vMtEntry 6 }


-- 1.0.62379.7.1.3.1.1.7
vMtVideoBitRateType OBJECT-TYPE
   SYNTAX BitRateType
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current video bit rate type applied.

      unspecified (0),
      Variable Bit Rate - VBR (1),
      Constant Bite Rate - CBR (2)"
   ::= { vMtEntry 7 }


-- 1.0.62379.7.1.3.1.1.8
vMtVideoBitRate OBJECT-TYPE
   SYNTAX CardinalNumber
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current video bit rate in kbps.

      If vMtVideoBitRateType = Unspecified (0), then
      this object shall be set to a default value of Unspecified (0)

      If vMtVideoBitRateType = VBR (1),
```

```
      then this object shall be the advertised Maximum Bit Rate

      If vMtVideoBitRateType = CBR (2),
      then this object shall be the advertised (Constant) Bit Rate"
   ::= { vMtEntry 8 }


-- 1.0.62379.7.1.3.1.1.9
vMtAspectRatio OBJECT-TYPE
   SYNTAX MediaFormat
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the current video aspect ratio.

      The format is defined in IEC62397-3 Ed.1
      It has the following format:

      1.0.62379.3.2.1.5.y.z where

      y is the source aspect ratio
      4:3  (43),
      16:9 (169),
      2.21:1 (221)

      z is the active format description code for the source aspect ratio.
      The codes are from 0000-1111
      See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { vMtEntry 9 }


-- 1.0.62379.7.1.3.1.1.10
vMtFECType OBJECT-TYPE
   SYNTAX VideoFECType
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "An indication as to the FEC type applied, if present.

      Current Values are:

      none(0),
      smpte2022(2),
      proprietary(4)"
   ::= { vMtEntry 10 }


-- 1.0.62379.7.1.3.1.1.11
vMtFECLengthDimension OBJECT-TYPE
   SYNTAX IntegerNumber
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the number of bytes over which FEC is
      applied and the matrix size being used. If the value of
      vMtFECType is zero (no FEC present), the value of zero
      shall be returned.
```

```
            The format shall be xxyy, where, xx represents the number
            of bytes over which FEC is applied and yy represents the
            matrix size, for example, 2405 for 24 bytes and a 5x5 matrix."
         ::= { vMtEntry 11 }


      -- 1.0.62379.7.1.3.1.1.12
      vMtTrickModeSupport OBJECT-TYPE
         SYNTAX TruthValue
         MAX-ACCESS read-only
         STATUS current
         DESCRIPTION
            "An indication as to whether trick mode is supported or not."
         ::= { vMtEntry 12 }


      -- 1.0.62379.7.1.3.2
      videoMeasurementGroup OBJECT-GROUP
         OBJECTS { vMtAudioBlockId, vMtNetworkBlockId, vMtVideoStatus, vMtVideoSourceFormat,
vMtVideoCodingType,
            vMtVideoBitRateType, vMtVideoBitRate, vMtAspectRatio, vMtFECType,
vMtFECLengthDimension,
            vMtTrickModeSupport }
         STATUS current
         DESCRIPTION
            "The group of objects used for video measurements."
         ::= { videoMeasurement 2 }


      -- 1.0.62379.7.1.4
      receiverMeasurement OBJECT IDENTIFIER ::= { measurementMIB 4 }


      -- 1.0.62379.7.1.4.2
      rxPointTable OBJECT-TYPE
         SYNTAX SEQUENCE OF RxPointEntry
         MAX-ACCESS not-accessible
         STATUS current
         DESCRIPTION
            "Entries in this table shall only exist if the value of nMtTxRxPoint
            is TRUE in the corresponding row in the nMtTable.

            The number of entries is between zero and the number of entries in the
            nMtTable. Since an entry in the nMtTable may not have the value of
            TRUE for nMtTxRxPoint, there may not be an entry in this
            table for each entry in the nMtTable.

            In addition to the index of rxPointBlockId, this table also uses the
            same index of nMtBlockId as the nMtTable."
         ::= { receiverMeasurement 2 }


      -- 1.0.62379.7.1.4.2.1
      rxPointEntry OBJECT-TYPE
         SYNTAX RxPointEntry
         MAX-ACCESS not-accessible
```

```
   STATUS current
   DESCRIPTION
      "A row in the receiver point table. Entries cannot
      be created or deleted with SNMP operations."
   INDEX { rxPointBlockId, rxPointNetworkBlockId }
   ::= { rxPointTable 1 }


RxPointEntry ::=
   SEQUENCE {
      rxPointBlockId
         BlockId,
      rxPointNetworkBlockId
         BlockId,
      rxPointBufferSize
         BufferSize,
      rxPointBufferOcpancyTime
         BufferOcpncyTime,
      rxPointBufferOcpncyPcnt
         BufferOcpncyPercent,
      rxPointMDI
         Utf8String,
      rxPointTSDF
         CardinalNumber
    }

-- 1.0.62379.7.1.4.2.1.1
rxPointBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The block identifier for this block.
      Used as an index when accessing the receiver point
      measurement block table."
   ::= { rxPointEntry 1 }


-- 1.0.62379.7.1.4.2.1.2
rxPointNetworkBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The index value (nMtBlockId) from the Network Measurement
      table (nMtTable) used as an index when accessing the rxPoint
      measurement block table."
   ::= { rxPointEntry 2 }


-- 1.0.62379.7.1.4.2.1.3
rxPointBufferSize OBJECT-TYPE
   SYNTAX BufferSize
   UNITS "ms"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
```

```
      "A description of the current total size of the receive buffer
      in ms."
   ::= { rxPointEntry 3 }


-- 1.0.62379.7.1.4.2.1.4
rxPointBufferOcpancyTime OBJECT-TYPE
   SYNTAX BufferOcpncyTime
   UNITS "ms"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the amount of data, in ms, occupying the
      receive buffer.
      This ranges between zero and rxPointBufferSize."
   ::= { rxPointEntry 4 }


-- 1.0.62379.7.1.4.2.1.5
rxPointBufferOcpncyPcnt OBJECT-TYPE
   SYNTAX BufferOcpncyPercent
   MAX-ACCESS read-write
   STATUS current
   DESCRIPTION
      "A description of the amount of data, expressed as a
      percentage of the total receive buffer size, rxPointBufferSize,
      occupying the receive buffer."
   ::= { rxPointEntry 5 }


-- 1.0.62379.7.1.4.2.1.6
rxPointMDI OBJECT-TYPE
   SYNTAX Utf8String
   UNITS "ms"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the measured and calculated Media Delivery
      Index (MDI).
      The MDI has two components, the Delay Factor (DF) and the
      Media Loss Rate (MLR).
      MDI is expressed as simple pairs of absolute values in the
      form of DF:MLR, e.g. 06:10"
   ::= { rxPointEntry 6 }


-- 1.0.62379.7.1.4.2.1.7
rxPointTSDF OBJECT-TYPE
   SYNTAX CardinalNumber
   UNITS "ms"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A description of the Time-stamped Delay Factor (TS-DF).
      Time-stamped Delay Factor is calculated as:
      TS-DF = D(Max) - D(Min)
      For further details of this, see EBU TECH3337"
```

```
        ::= { rxPointEntry 7 }


      -- 1.0.62379.7.1.4.3
      receiverMeasurementGroup OBJECT-GROUP
        OBJECTS { rxPointBufferSize, rxPointBufferOcpancyTime, rxPointBufferOcpncyPcnt,
rxPointMDI, rxPointTSDF
            }
        STATUS current
        DESCRIPTION
          "The group of objects used for receiver measurements."
        ::= { receiverMeasurement 3 }


      -- 1.0.62379.7.1.5
      temperatureMeasurement OBJECT IDENTIFIER ::= { measurementMIB 5 }


      -- 1.0.62379.7.1.5.1
      temperatureTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TemperatureEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
          "Information describing the temperature (if applicable)
          measured within the unit at various locations (if applicable)."
        ::= { temperatureMeasurement 1 }


      -- 1.0.62379.7.1.5.1.1
      temperatureEntry OBJECT-TYPE
        SYNTAX TemperatureEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
          "Description."
        INDEX { temperatureBlockId, temperatureLocnNumber }
        ::= { temperatureTable 1 }


      TemperatureEntry ::=
        SEQUENCE {
          temperatureBlockId
            BlockId,
          temperatureLocnNumber
            IndexNumber,
          temperatureLocation
            TemperatureLocn,
          temperatureTrend
            TemperatureTrend,
          temperatureStatus
            TemperatureStatus,
          temperatureLowWarning
            IntegerNumber,
          temperatureHighWarning
            IntegerNumber,
          temperatureLowCritical
```

```
        IntegerNumber,
      temperatureHighCritical
        IntegerNumber
    }

-- 1.0.62379.7.1.5.1.1.1
temperatureBlockId OBJECT-TYPE
   SYNTAX BlockId
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The block identifier for this block.
      Used as an index when accessing the temperature
      measurement block table."
   ::= { temperatureEntry 1 }


-- 1.0.62379.7.1.5.1.1.2
temperatureLocnNumber OBJECT-TYPE
   SYNTAX IndexNumber
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
      "The identifier where the temperature is measured.
      Used as an index when accessing the unit temperature
      measurement table.
      Each temperature measurement point in a unit has a unique
      number.
      Temperature measurement location numbers should be
      allocated sequentially, starting from 1."
   ::= { temperatureEntry 2 }


-- 1.0.62379.7.1.5.1.1.3
temperatureLocation OBJECT-TYPE
   SYNTAX TemperatureLocn
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "A text description of the location of where the temperature is
      being measured."
   ::= { temperatureEntry 3 }


-- 1.0.62379.7.1.5.1.1.4
temperatureTrend OBJECT-TYPE
   SYNTAX TemperatureTrend
   UNITS "Degrees Celsius"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The current temperature measured at this location.
      Unit is Degrees Celsius"
   ::= { temperatureEntry 4 }


-- 1.0.62379.7.1.5.1.1.5
```

```
temperatureStatus OBJECT-TYPE
   SYNTAX TemperatureStatus
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The status of this particular temperature sensor.

      Current Values are:

      undetermined(0),
      other(1),
      unknown(2),
      ok(3),
      warning(4),
      critical(5),
      non-recoverable(6)"
   ::= { temperatureEntry 5 }


-- 1.0.62379.7.1.5.1.1.6
temperatureLowWarning OBJECT-TYPE
   SYNTAX IntegerNumber
   UNITS "Degrees Celcius"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The low warning limit for the measured temperature.
      Unit is Degrees Celsius."
   ::= { temperatureEntry 6 }


-- 1.0.62379.7.1.5.1.1.7
temperatureHighWarning OBJECT-TYPE
   SYNTAX IntegerNumber
   UNITS "Degrees Celcius"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The high warning limit for the measured temperature.
      Unit is Degrees Celsius."
   ::= { temperatureEntry 7 }


-- 1.0.62379.7.1.5.1.1.8
temperatureLowCritical OBJECT-TYPE
   SYNTAX IntegerNumber
   UNITS "Degrees Celcius"
   MAX-ACCESS read-only
   STATUS current
   DESCRIPTION
      "The low critical limit for the measured temperature.
      Unit is Degrees Celsius."
   ::= { temperatureEntry 8 }


-- 1.0.62379.7.1.5.1.1.9
temperatureHighCritical OBJECT-TYPE
```

```
        SYNTAX IntegerNumber
        UNITS "Degrees Celcius"
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "The high critical limit for the measured temperature.
           Unit is Degrees Celsius."
        ::= { temperatureEntry 9 }


     -- 1.0.62379.7.1.5.2
     temperatureMeasurementGroup OBJECT-GROUP
        OBJECTS { temperatureLocation, temperatureTrend, temperatureStatus,
temperatureLowWarning, temperatureHighWarning,
           temperatureLowCritical, temperatureHighCritical }
        STATUS current
        DESCRIPTION
           "The group of objects used for temperature measurements."
        ::= { temperatureMeasurement 2 }



   END
```

Annex D: (Informative) Machine-readable textual conventions definitions

This annex provides a machine-readable version of the textual conventions definitions specific to this standard which is intended to be interpretable by standard MIB browsing software tools. If there is any inconsistency between this annex and the appropriate clauses in IEC 62379-7, the appropriate clauses in IEC 62379-7 take precedence.

The format used to describe the textual conventions definitions conforms to IETF STD 58 (SMIv2).

```
   IEC62379-7-IPM-TC-MIB DEFINITIONS ::= BEGIN

      IMPORTS
         iec62379
            FROM IEC62379-1-MIB
         Unsigned32, Gauge32, MODULE-IDENTITY
            FROM SNMPv2-SMI
         TEXTUAL-CONVENTION
            FROM SNMPv2-TC;


      -- 1.0.62379.7.3
      measurementTCMIB MODULE-IDENTITY
         LAST-UPDATED "201106211200Z"    -- June 21, 2011 at 12:00 GMT
         ORGANIZATION
            "IEC PT62379"
         CONTACT-INFO
            "Not specified."
         DESCRIPTION
            "The Textual Conventions MIB module for managing
            measurement functions in IEC 62379 compliant
            equipment.
            The contents of this MIB Module have been defined by
            the EBU ECN-IPM (European Broadcasting Union
            Expert Communities Networks, IP Measurements group)."
         REVISION "201106101200Z"     -- June 10, 2011 at 12:00 GMT
         DESCRIPTION
            "Added BitRateType textual convention."
         ::= { measurement 3 }



--
-- Textual conventions
--

      NetworkType ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An enumeration identifying a network type of over which the
            media is flowing."
         SYNTAX INTEGER
            {
```

```
          ipv4(1),
          ipv6(2),
          asi(3)
          }
-- {
-- ipv4(1),
-- ipv6(2),
-- asi(3)
-- } (ipv4..asi)
--
     TransportType ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
          "An enumeration identifying a transport type of over which the
          media is flowing.

          Note that the values for this textual convention are NOT the same
          as the numbers used in the protocol field of IPv4 packets and
          the Next Header Field of IPv6 packets.
          See http://www.iana.org/assignments/protocol-numbers"
        SYNTAX INTEGER
          {
          notApplicable(0),
          rtp(1)
          }
-- {
-- notApplicable(0),
-- rtp(1)
-- }(notApplicable..rtp)
--
     AudioFECType ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
          "An enumeration identifying the FEC type applied, if present."
        SYNTAX INTEGER
          {
          none(0),
          smpte2021(1),
          smpte2022(2),
          rfc2733(3),
          proprietary(4)
          }
-- {
-- none(0),
-- smpte2021(1),
-- smpte2022(2),
-- rfc2733(3),
-- proprietary(4)
-- } (none..proprietary)
--
     VideoFECType ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
          "An enumeration identifying the FEC type applied, if present."
        SYNTAX INTEGER
          {
          none(0),
```

```
            smpte2022(2),
            proprietary(4)
            }
-- {
-- none(0),
-- smpte2022(2),
-- proprietary(4)
-- } (none..proprietary)
--
      BufferSize ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "A type to indicate the current total size of the receive buffer in
            ms."
         SYNTAX Unsigned32

      BufferOcpncyTime ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "A type to report the amount of data, expressed in ms,
            occupying the receive buffer."
         SYNTAX Gauge32

      BufferOcpncyPercent ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "A type to report the amount of data, expressed as a
            percentage of the total receive buffer size, occupying the
            receive buffer."
         SYNTAX INTEGER (1..100)

      TemperatureLocn ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "A type to indicate the location where the temperature is measured."
         SYNTAX OCTET STRING (SIZE (0..80))

      TemperatureTrend ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "A type to report the current temperature at the measured
            location.
            The use of this type will allow changes (either up or down)
            to be reported."
         SYNTAX Gauge32

      TemperatureStatus ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An enumeration identifying the temperature status levels.
            Semantics are equipment specific."
         SYNTAX INTEGER
            {
            undetermined(0),
            other(1),
            unknown(2),
            ok(3),
```

```
          warning(4),
          critical(5),
          nonRecoverable(6)
          }
-- {
-- undetermined(0),
-- other(1),
-- unknown(2),
-- ok(3),
-- warning(4),
-- critical(5),
-- nonRecoverable(6)
-- } (undetermined..nonRecoverable)
--
     BitRateType ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An enumeration identifying the video bit rate type applied

           vbr = variable bit rate
           cbr = constant bit rate"
        SYNTAX INTEGER
           {
           unspecified(0),
           vbr(1),
           cbr(2)
           }
-- {
-- unspecified(0),
-- vbr(1),
-- cbr(2)
-- } (unspecified..cbr)
--

--
-- Node definitions
--

     -- 1.0.62379.7
     measurement OBJECT IDENTIFIER ::= { iec62379 7 }



   END
```

Annex E: (Informative) Machine-readable audio format definitions

This annex provides a machine-readable version of the audio data format definitions specific to this standard which is intended to be interpretable by standard MIB browsing software tools. If there is any inconsistency between this annex and clause 4 of IEC 62379-2, clause 4 of IEC 62379-2 takes precedence.

The format used to describe the textual conventions definitions conforms to IETF STD 58 (SMIv2).

Note 1: This annex is not intended to cover every format permitted by the definitions in clause 4 of IEC 62379-2.

Note 2: This annex does not currently include the formats in clause 2.4.2.8 for MPEG-4 HE-AACv2 or clause 2.4.2.9 for MPEG-4 AAC-ELD of this document. These formats and any extensions for the other associated audio format parameters, such as channel arrangement, number of channels, etc., may be included (in the aacAudio section - 1.0.62379.2.2.1.6 following aacLD - 1.0.62379.2.2.1.6.5) following the same style as other fully qualified audio formats shown. It is expected that an amendment will be made to IEC 62379-2 to include these and any other new formats in the future.

```
IEC62379-2-FORMATS DEFINITIONS ::= BEGIN

   IMPORTS
      iec62379
         FROM IEC62379-1-MIB
      MODULE-IDENTITY, OBJECT-IDENTITY
         FROM SNMPv2-SMI
      TEXTUAL-CONVENTION
         FROM SNMPv2-TC;


   -- 1.0.62379.2.2
   audioFormat MODULE-IDENTITY
      LAST-UPDATED "200804301500Z"    -- April 30, 2008 at 15:00 GMT
      ORGANIZATION
         "IEC PT62379"
      CONTACT-INFO
         "Not specified"
      DESCRIPTION
         "The audio format identifiers defined in clause 4 of
         IEC 62379-2."
      REVISION "200804301500Z"     -- April 30, 2008 at 15:00 GMT
      DESCRIPTION
         "Edition 1.0 final draft."
      ::= { audio 2 }



--
-- Textual conventions
--
```

```
--  4.1 Audio signal format definitions
-- 4.1.1 Audio parameters
     ChannelArrangement ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "The audio channel arrangement in the audio format."
        SYNTAX INTEGER (0..5)

--  {
-- unspecified         (0),
-- discreteMono        (1),
-- stereo              (2),
-- jointStereo         (3),
-- surround            (4),
-- surroundWithDownmix (5)
-- } (unspecified..surroundWithDownmix)
     NumberChannels ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An integer representing the number of audio channels."
        SYNTAX INTEGER (0..2147483647)

     BitDepth ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An integer representing the audio bit depth in bits per
           sample."
        SYNTAX INTEGER (0..2147483647)

     SamplingFrequency ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An integer representing the audio sample frequency in Hz."
        SYNTAX INTEGER (0..2147483647)

     BitRate ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An integer representing the audio bit rate in bits per second."
        SYNTAX INTEGER (0..2147483647)


--
-- Node definitions
--

     -- 1.0.62379.2
     audio OBJECT IDENTIFIER ::= { iec62379 2 }


--  4.1.2 Audio signal formats
     -- 1.0.62379.2.2.1
     audioSignalFormat OBJECT IDENTIFIER ::= { audioFormat 1 }


--  4.1.2.1 Unspecified audio
     -- 1.0.62379.2.2.1.0
```

```
      unspecifiedAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "AudioFormat wildcard value - any supported format allowed."
         ::= { audioSignalFormat 0 }


--  4.1.2.2 No audio
      -- 1.0.62379.2.2.1.1
      noAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Indicates the output is muted."
         ::= { audioSignalFormat 1 }


--  4.1.2.3 Analogue audio
      -- 1.0.62379.2.2.1.2
      analogueAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Analogue audio."
         ::= { audioSignalFormat 2 }


--  analogue discrete mono
      -- 1.0.62379.2.2.1.2.1
      analogueMono OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono analogue audio."
         ::= { analogueAudio 1 }


      -- 1.0.62379.2.2.1.2.1.1
      analogueMono1Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono analogue audio with 1 channel."
         ::= { analogueMono 1 }


      -- 1.0.62379.2.2.1.2.1.2
      analogueMono2Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono analogue audio with 2 channels."
         ::= { analogueMono 2 }


      -- 1.0.62379.2.2.1.2.1.3
      analogueMono3Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono analogue audio with 3 channels."
         ::= { analogueMono 3 }
```

```
      -- 1.0.62379.2.2.1.2.1.4
      analogueMono4Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono analogue audio with 4 channels."
         ::= { analogueMono 4 }


--  analogue stereo
      -- 1.0.62379.2.2.1.2.2
      analogueStereo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Stereo analogue audio."
         ::= { analogueAudio 2 }


      -- 1.0.62379.2.2.1.2.2.2
      analogueStereo2Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Stereo analogue audio with 2 channels (1 L-R pair)."
         ::= { analogueStereo 2 }


      -- 1.0.62379.2.2.1.2.2.4
      analogueStereo4Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Stereo analogue audio with 4 channels (2 L-R pairs)."
         ::= { analogueStereo 4 }


--  analogue jointstereo
      -- 1.0.62379.2.2.1.2.3
      analogueJointStereo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Joint stereo analogue audio."
         ::= { analogueAudio 3 }


      -- 1.0.62379.2.2.1.2.3.2
      analogueJointStereo2Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Joint stereo analogue audio with 2 channels (1 M-S pair)."
         ::= { analogueJointStereo 2 }


      -- 1.0.62379.2.2.1.2.3.4
      analogueJointStereo4Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Joint stereo analogue audio with 4 channels (2 M-S pairs)."
         ::= { analogueJointStereo 4 }
```

```
--  4.1.2.4 PCM audio
     -- 1.0.62379.2.2.1.3
     pcmAudio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Linear PCM audio."
        ::= { audioSignalFormat 3 }


--  PCM discrete mono
     -- 1.0.62379.2.2.1.3.1
     pcmMono OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Discrete mono linear PCM audio."
        ::= { pcmAudio 1 }


     -- 1.0.62379.2.2.1.3.1.1
     pcmMono1Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Discrete mono linear PCM audio with 1 channel."
        ::= { pcmMono 1 }


     -- 1.0.62379.2.2.1.3.1.1.16
     pcmMono1Chan16 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "16-bit discrete mono linear PCM audio with 1 channel."
        ::= { pcmMono1Chan 16 }


     -- 1.0.62379.2.2.1.3.1.1.16.44100
     pcmMono1Chan16at44100 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "44.1k 16-bit discrete mono linear PCM audio with 1 channel."
        ::= { pcmMono1Chan16 44100 }


     -- 1.0.62379.2.2.1.3.1.1.16.48000
     pcmMono1Chan16at48000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "48k 16-bit discrete mono linear PCM audio with 1 channel."
        ::= { pcmMono1Chan16 48000 }


     -- 1.0.62379.2.2.1.3.1.1.16.96000
     pcmMono1Chan16at96000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "96k 16-bit discrete mono linear PCM audio with 1 channel."
```

```
   ::= { pcmMono1Chan16 96000 }


-- 1.0.62379.2.2.1.3.1.1.20
pcmMono1Chan20 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "20-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan 20 }


-- 1.0.62379.2.2.1.3.1.1.20.44100
pcmMono1Chan20at44100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k 20-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan20 44100 }


-- 1.0.62379.2.2.1.3.1.1.20.48000
pcmMono1Chan20at48000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k 20-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan20 48000 }


-- 1.0.62379.2.2.1.3.1.1.20.96000
pcmMono1Chan20at96000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "96k 20-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan20 96000 }


-- 1.0.62379.2.2.1.3.1.1.24
pcmMono1Chan24 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "24-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan 24 }


-- 1.0.62379.2.2.1.3.1.1.24.44100
pcmMono1Chan24at44100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k 24-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan24 44100 }


-- 1.0.62379.2.2.1.3.1.1.24.48000
pcmMono1Chan24at48000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k 24-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan24 48000 }
```

```
-- 1.0.62379.2.2.1.3.1.1.24.96000
pcmMono1Chan24at96000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "96k 24-bit discrete mono linear PCM audio with 1 channel."
   ::= { pcmMono1Chan24 96000 }


-- 1.0.62379.2.2.1.3.1.2
pcmMono2Chan OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "Discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono 2 }


-- 1.0.62379.2.2.1.3.1.2.16
pcmMono2Chan16 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "16-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan 16 }


-- 1.0.62379.2.2.1.3.1.2.16.44100
pcmMono2Chan16at44100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k 16-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan16 44100 }


-- 1.0.62379.2.2.1.3.1.2.16.48000
pcmMono2Chan16at48000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k 16-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan16 48000 }


-- 1.0.62379.2.2.1.3.1.2.16.96000
pcmMono2Chan16at96000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "96k 16-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan16 96000 }


-- 1.0.62379.2.2.1.3.1.2.20
pcmMono2Chan20 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "20-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan 20 }
```

```
-- 1.0.62379.2.2.1.3.1.2.20.44100
pcmMono2Chan20at44100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k 20-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan20 44100 }


-- 1.0.62379.2.2.1.3.1.2.20.48000
pcmMono2Chan20at48000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k 20-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan20 48000 }


-- 1.0.62379.2.2.1.3.1.2.20.96000
pcmMono2Chan20at96000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "96k 20-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan20 96000 }


-- 1.0.62379.2.2.1.3.1.2.24
pcmMono2Chan24 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "24-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan 24 }


-- 1.0.62379.2.2.1.3.1.2.24.44100
pcmMono2Chan24at44100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k 24-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan24 44100 }


-- 1.0.62379.2.2.1.3.1.2.24.48000
pcmMono2Chan24at48000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k 24-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan24 48000 }


-- 1.0.62379.2.2.1.3.1.2.24.96000
pcmMono2Chan24at96000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "96k 24-bit discrete mono linear PCM audio with 2 channels."
   ::= { pcmMono2Chan24 96000 }
```

```
--  PCM stereo
     -- 1.0.62379.2.2.1.3.2
     pcmStereo OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Stereo linear PCM audio."
        ::= { pcmAudio 2 }


     -- 1.0.62379.2.2.1.3.2.2
     pcmStereo2Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Stereo linear PCM audio with 2 channels (1 L-R pair)."
        ::= { pcmStereo 2 }


     -- 1.0.62379.2.2.1.3.2.2.16
     pcmStereo2Chan16 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "16-bit stereo linear PCM audio with 2 channels."
        ::= { pcmStereo2Chan 16 }


     -- 1.0.62379.2.2.1.3.2.2.16.44100
     pcmStereo2Chan16at44100 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "44.1k 16-bit stereo linear PCM audio with 2 channels."
        ::= { pcmStereo2Chan16 44100 }


     -- 1.0.62379.2.2.1.3.2.2.16.48000
     pcmStereo2Chan16at48000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "48k 16-bit stereo linear PCM audio with 2 channels."
        ::= { pcmStereo2Chan16 48000 }


     -- 1.0.62379.2.2.1.3.2.2.16.96000
     pcmStereo2Chan16at96000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "96k 16-bit stereo linear PCM audio with 2 channels."
        ::= { pcmStereo2Chan16 96000 }


     -- 1.0.62379.2.2.1.3.2.2.20
     pcmStereo2Chan20 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "20-bit stereo linear PCM audio with 2 channels."
        ::= { pcmStereo2Chan 20 }
```

```
   -- 1.0.62379.2.2.1.3.2.2.20.44100
   pcmStereo2Chan20at44100 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "44.1k 20-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan20 44100 }


   -- 1.0.62379.2.2.1.3.2.2.20.48000
   pcmStereo2Chan20at48000 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "48k 20-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan20 48000 }


   -- 1.0.62379.2.2.1.3.2.2.20.96000
   pcmStereo2Chan20at96000 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "96k 20-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan20 96000 }


   -- 1.0.62379.2.2.1.3.2.2.24
   pcmStereo2Chan24 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "24-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan 24 }


   -- 1.0.62379.2.2.1.3.2.2.24.44100
   pcmStereo2Chan24at44100 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "44.1k 24-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan24 44100 }


   -- 1.0.62379.2.2.1.3.2.2.24.48000
   pcmStereo2Chan24at48000 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "48k 24-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan24 48000 }


   -- 1.0.62379.2.2.1.3.2.2.24.96000
   pcmStereo2Chan24at96000 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "96k 24-bit stereo linear PCM audio with 2 channels."
      ::= { pcmStereo2Chan24 96000 }


--  PCM joint stereo
```

```
      -- 1.0.62379.2.2.1.3.3
      pcmJointStereo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Joint stereo linear PCM audio."
         ::= { pcmAudio 3 }


      -- 1.0.62379.2.2.1.3.3.2
      pcmJointStereo2Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Joint stereo linear PCM audio with 2 channels (1 M-S pair)."
         ::= { pcmJointStereo 2 }


      -- 1.0.62379.2.2.1.3.3.2.16
      pcmJointStereo2Chan16 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "16-bit joint stereo linear PCM audio with 2 channels."
         ::= { pcmJointStereo2Chan 16 }


      -- 1.0.62379.2.2.1.3.3.2.20
      pcmJointStereo2Chan20 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "20-bit joint stereo linear PCM audio with 2 channels."
         ::= { pcmJointStereo2Chan 20 }


      -- 1.0.62379.2.2.1.3.3.2.24
      pcmJointStereo2Chan24 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "24-bit joint stereo linear PCM audio with 2 channels."
         ::= { pcmJointStereo2Chan 24 }


--  4.1.2.5 MPEG-1 layer 2 audio
      -- 1.0.62379.2.2.1.4
      mp2Audio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "MPEG1 layer 2 audio."
         ::= { audioSignalFormat 4 }


--  MP2 discrete mono
      -- 1.0.62379.2.2.1.4.1
      mp2Mono OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono MP2 audio."
         ::= { mp2Audio 1 }
```

```
      -- 1.0.62379.2.2.1.4.1.1
      mp2Mono1Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Discrete mono MP2 audio with 1 channel."
         ::= { mp2Mono 1 }


--  MP2 discrete mono with 1 channel at 44100
      -- 1.0.62379.2.2.1.4.1.1.44100
      mp2Mono1Chan44100 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio with 1 channel."
         ::= { mp2Mono1Chan 44100 }


      -- 1.0.62379.2.2.1.4.1.1.44100.32000
      mp2Mono1Chan44100at32 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 32kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 32000 }


      -- 1.0.62379.2.2.1.4.1.1.44100.40000
      mp2Mono1Chan44100at40 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 40kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 40000 }


      -- 1.0.62379.2.2.1.4.1.1.44100.48000
      mp2Mono1Chan44100at48 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 48kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 48000 }


      -- 1.0.62379.2.2.1.4.1.1.44100.56000
      mp2Mono1Chan44100at56 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 56kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 56000 }


      -- 1.0.62379.2.2.1.4.1.1.44100.64000
      mp2Mono1Chan44100at64 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 64kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 64000 }
```

```
-- 1.0.62379.2.2.1.4.1.1.44100.80000
mp2Mono1Chan44100at80 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 80kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 80000 }


-- 1.0.62379.2.2.1.4.1.1.44100.96000
mp2Mono1Chan44100at96 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 96kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 96000 }


-- 1.0.62379.2.2.1.4.1.1.44100.112000
mp2Mono1Chan44100at112 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 112kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 112000 }


-- 1.0.62379.2.2.1.4.1.1.44100.128000
mp2Mono1Chan44100at128 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 128kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 128000 }


-- 1.0.62379.2.2.1.4.1.1.44100.160000
mp2Mono1Chan44100at160 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 160kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 160000 }


-- 1.0.62379.2.2.1.4.1.1.44100.192000
mp2Mono1Chan44100at192 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 192kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 192000 }


-- 1.0.62379.2.2.1.4.1.1.44100.224000
mp2Mono1Chan44100at224 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k discrete mono MP2 audio at 224kb/s with 1 channel."
   ::= { mp2Mono1Chan44100 224000 }
```

```
      -- 1.0.62379.2.2.1.4.1.1.44100.256000
      mp2Mono1Chan44100at256 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 256kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 256000 }


      -- 1.0.62379.2.2.1.4.1.1.44100.320000
      mp2Mono1Chan44100at320 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k discrete mono MP2 audio at 320kb/s with 1 channel."
         ::= { mp2Mono1Chan44100 320000 }


--  MP2 discrete mono with 1 channel at 48000
      -- 1.0.62379.2.2.1.4.1.1.48000
      mp2Mono1Chan48000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k mono MP2 audio with 1 channel."
         ::= { mp2Mono1Chan 48000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.32000
      mp2Mono1Chan48000at32 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k discrete mono MP2 audio at 32kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 32000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.40000
      mp2Mono1Chan48000at40 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "8k discrete mono MP2 audio at 40kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 40000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.48000
      mp2Mono1Chan48000at48 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k discrete mono MP2 audio at 48kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 48000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.56000
      mp2Mono1Chan48000at56 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k discrete mono MP2 audio at 56kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 56000 }
```

```
-- 1.0.62379.2.2.1.4.1.1.48000.64000
mp2Mono1Chan48000at64 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 64kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 64000 }


-- 1.0.62379.2.2.1.4.1.1.48000.80000
mp2Mono1Chan48000at80 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 80kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 80000 }


-- 1.0.62379.2.2.1.4.1.1.48000.96000
mp2Mono1Chan48000at96 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 96kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 96000 }


-- 1.0.62379.2.2.1.4.1.1.48000.112000
mp2Mono1Chan48000at112 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 112kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 112000 }


-- 1.0.62379.2.2.1.4.1.1.48000.128000
mp2Mono1Chan48000at128 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 128kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 128000 }


-- 1.0.62379.2.2.1.4.1.1.48000.160000
mp2Mono1Chan48000at160 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 160kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 160000 }


-- 1.0.62379.2.2.1.4.1.1.48000.192000
mp2Mono1Chan48000at192 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k discrete mono MP2 audio at 192kb/s with 1 channel."
   ::= { mp2Mono1Chan48000 192000 }


-- 1.0.62379.2.2.1.4.1.1.48000.224000
```

```
      mp2Mono1Chan48000at224 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k discrete mono MP2 audio at 224kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 224000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.256000
      mp2Mono1Chan48000at256 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k discrete mono MP2 audio at 256kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 256000 }


      -- 1.0.62379.2.2.1.4.1.1.48000.320000
      mp2Mono1Chan48000at320 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "8k discrete mono MP2 audio at 320kb/s with 1 channel."
         ::= { mp2Mono1Chan48000 320000 }


--  MP2 stereo
      -- 1.0.62379.2.2.1.4.2
      mp2Stereo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Stereo MP2 audio."
         ::= { mp2Audio 2 }


      -- 1.0.62379.2.2.1.4.2.2
      mp2Stereo2Chan OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Stereo MP2 audio with 2 channels (1 L-R pair)."
         ::= { mp2Stereo 2 }


--  MP2 stereo with 2 channels at 44100
      -- 1.0.62379.2.2.1.4.2.2.44100
      mp2Stereo2Chan44100 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio."
         ::= { mp2Stereo2Chan 44100 }


      -- 1.0.62379.2.2.1.4.2.2.44100.32000
      mp2Stereo2Chan44100at32 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 32kb/s."
         ::= { mp2Stereo2Chan44100 32000 }
```

```
-- 1.0.62379.2.2.1.4.2.2.44100.40000
mp2Stereo2Chan44100at40 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 40kb/s."
   ::= { mp2Stereo2Chan44100 40000 }


-- 1.0.62379.2.2.1.4.2.2.44100.48000
mp2Stereo2Chan44100at48 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 48kb/s."
   ::= { mp2Stereo2Chan44100 48000 }


-- 1.0.62379.2.2.1.4.2.2.44100.56000
mp2Stereo2Chan44100at56 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 56kb/s."
   ::= { mp2Stereo2Chan44100 56000 }


-- 1.0.62379.2.2.1.4.2.2.44100.64000
mp2Stereo2Chan44100at64 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 64kb/s."
   ::= { mp2Stereo2Chan44100 64000 }


-- 1.0.62379.2.2.1.4.2.2.44100.80000
mp2Stereo2Chan44100at80 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 80kb/s."
   ::= { mp2Stereo2Chan44100 80000 }


-- 1.0.62379.2.2.1.4.2.2.44100.96000
mp2Stereo2Chan44100at96 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 96kb/s."
   ::= { mp2Stereo2Chan44100 96000 }


-- 1.0.62379.2.2.1.4.2.2.44100.112000
mp2Stereo2Chan44100at112 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "44.1k stereo MP2 audio at 112kb/s."
   ::= { mp2Stereo2Chan44100 112000 }


-- 1.0.62379.2.2.1.4.2.2.44100.128000
```

```
      mp2Stereo2Chan44100at128 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 128kb/s."
         ::= { mp2Stereo2Chan44100 128000 }


      -- 1.0.62379.2.2.1.4.2.2.44100.160000
      mp2Stereo2Chan44100at160 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 160kb/s."
         ::= { mp2Stereo2Chan44100 160000 }


      -- 1.0.62379.2.2.1.4.2.2.44100.192000
      mp2Stereo2Chan44100at192 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 192kb/s."
         ::= { mp2Stereo2Chan44100 192000 }


      -- 1.0.62379.2.2.1.4.2.2.44100.224000
      mp2Stereo2Chan44100at224 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 224kb/s."
         ::= { mp2Stereo2Chan44100 224000 }


      -- 1.0.62379.2.2.1.4.2.2.44100.256000
      mp2Stereo2Chan44100at256 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 256kb/s."
         ::= { mp2Stereo2Chan44100 256000 }


      -- 1.0.62379.2.2.1.4.2.2.44100.320000
      mp2Stereo2Chan44100at320 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "44.1k stereo MP2 audio at 320kb/s."
         ::= { mp2Stereo2Chan44100 320000 }


--  MP2 stereo with 2 channels at 48000
      -- 1.0.62379.2.2.1.4.2.2.48000
      mp2Stereo2Chan48000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "48k stereo MP2 audio."
         ::= { mp2Stereo2Chan 48000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.32000
```

```
mp2Stereo2Chan48000at32 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 32kb/s."
   ::= { mp2Stereo2Chan48000 32000 }


-- 1.0.62379.2.2.1.4.2.2.48000.40000
mp2Stereo2Chan48000at40 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 40kb/s."
   ::= { mp2Stereo2Chan48000 40000 }


-- 1.0.62379.2.2.1.4.2.2.48000.48000
mp2Stereo2Chan48000at48 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 48kb/s."
   ::= { mp2Stereo2Chan48000 48000 }


-- 1.0.62379.2.2.1.4.2.2.48000.56000
mp2Stereo2Chan48000at56 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 56kb/s."
   ::= { mp2Stereo2Chan48000 56000 }


-- 1.0.62379.2.2.1.4.2.2.48000.64000
mp2Stereo2Chan48000at64 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 64kb/s."
   ::= { mp2Stereo2Chan48000 64000 }


-- 1.0.62379.2.2.1.4.2.2.48000.80000
mp2Stereo2Chan48000at80 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 80kb/s."
   ::= { mp2Stereo2Chan48000 80000 }


-- 1.0.62379.2.2.1.4.2.2.48000.96000
mp2Stereo2Chan48000at96 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "48k stereo MP2 audio at 96kb/s."
   ::= { mp2Stereo2Chan48000 96000 }


-- 1.0.62379.2.2.1.4.2.2.48000.112000
mp2Stereo2Chan48000at112 OBJECT-IDENTITY
```

```
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 112kb/s."
        ::= { mp2Stereo2Chan48000 112000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.128000
      mp2Stereo2Chan48000at128 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 128kb/s."
        ::= { mp2Stereo2Chan48000 128000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.160000
      mp2Stereo2Chan48000at160 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 160kb/s."
        ::= { mp2Stereo2Chan48000 160000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.192000
      mp2Stereo2Chan48000at192 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 192kb/s."
        ::= { mp2Stereo2Chan48000 192000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.224000
      mp2Stereo2Chan48000at224 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 224kb/s."
        ::= { mp2Stereo2Chan48000 224000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.256000
      mp2Stereo2Chan48000at256 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 256kb/s."
        ::= { mp2Stereo2Chan48000 256000 }


      -- 1.0.62379.2.2.1.4.2.2.48000.320000
      mp2Stereo2Chan48000at320 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "48k stereo MP2 audio at 320kb/s."
        ::= { mp2Stereo2Chan48000 320000 }


--  MP2 joint stereo
      -- 1.0.62379.2.2.1.4.3
      mp2JointStereo OBJECT-IDENTITY
```

```
        STATUS current
        DESCRIPTION
           "Joint stereo MP2 audio."
        ::= { mp2Audio 3 }


      -- 1.0.62379.2.2.1.4.3.2
      mp2JointStereo2Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Joint stereo MP2 audio with 2 channels (1 M-S pair)."
        ::= { mp2JointStereo 2 }


      -- 1.0.62379.2.2.1.4.3.2.44100
      mp2JointStereo2Chan44100 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "44.1k joint stereo MP2 audio."
        ::= { mp2JointStereo2Chan 44100 }


      -- 1.0.62379.2.2.1.4.3.2.48000
      mp2JointStereo2Chan48000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "48k joint stereo MP2 audio."
        ::= { mp2JointStereo2Chan 48000 }


--  4.1.2.6 MPEG-1 layer 3 audio
      -- 1.0.62379.2.2.1.5
      mp3Audio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "MPEG1 layer 3 audio."
        ::= { audioSignalFormat 5 }


      -- 1.0.62379.2.2.1.5.1
      mp3Mono OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Discrete mono MP3 audio."
        ::= { mp3Audio 1 }


      -- 1.0.62379.2.2.1.5.1.1
      mp3Mono1Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Discrete mono MP3 audio with 1 channel."
        ::= { mp3Mono 1 }


      -- 1.0.62379.2.2.1.5.2
      mp3Stereo OBJECT-IDENTITY
```

```
        STATUS current
        DESCRIPTION
          "Stereo MP3 audio."
        ::= { mp3Audio 2 }


      -- 1.0.62379.2.2.1.5.2.2
      mp3Stereo2Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "Stereo MP3 audio with 2 channels (1 L-R pair)."
        ::= { mp3Stereo 2 }


      -- 1.0.62379.2.2.1.5.3
      mp3JointStereo OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "Joint stereo MP3 audio."
        ::= { mp3Audio 3 }


      -- 1.0.62379.2.2.1.5.3.2
      mp3JointStereo2Chan OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "Joint stereo MP3 audio with 2 channels (1 M-S pair)."
        ::= { mp3JointStereo 2 }


--  4.1.2.7 AAC audio
      -- 1.0.62379.2.2.1.6
      aacAudio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "aac audio."
        ::= { audioSignalFormat 6 }


      -- 1.0.62379.2.2.1.6.1
      aacLC OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "aac audio with the low complexity profile."
        ::= { aacAudio 1 }


      -- 1.0.62379.2.2.1.6.2
      aacMain OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "aac audio with the main profile."
        ::= { aacAudio 2 }


      -- 1.0.62379.2.2.1.6.3
      aacSRS OBJECT-IDENTITY
```

```
        STATUS current
        DESCRIPTION
           "aac audio with the sample-rate-scalable profile."
        ::= { aacAudio 3 }


      -- 1.0.62379.2.2.1.6.4
      aacLTP OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "aac audio with the long term prediction profile."
        ::= { aacAudio 4 }


      -- 1.0.62379.2.2.1.6.5
      aacLD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "aac audio with the low delay profile."
        ::= { aacAudio 5 }


--  4.1.2.8 G711 audio
      -- 1.0.62379.2.2.1.7
      g711Audio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "G711 audio."
        ::= { audioSignalFormat 7 }


      -- 1.0.62379.2.2.1.7.1
      g711ALaw OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "G711 audio denoting A-Law encoding."
        ::= { g711Audio 1 }


      -- 1.0.62379.2.2.1.7.2
      g711MuLaw OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "G711 audio denoting mu-Law encoding."
        ::= { g711Audio 2 }


--  4.1.2.9 G722 audio
      -- 1.0.62379.2.2.1.8
      g722Audio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "G722 audio."
        ::= { audioSignalFormat 8 }


      -- 1.0.62379.2.2.1.8.48
```

```
      g722at48 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "G722 audio at 48kbits/s."
         ::= { g722Audio 48 }


      -- 1.0.62379.2.2.1.8.56
      g722at56 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "G722 audio at 56kbits/s."
         ::= { g722Audio 56 }


      -- 1.0.62379.2.2.1.8.64
      g722at64 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "G722 audio at 64kbits/s."
         ::= { g722Audio 64 }


--  4.1.2.10 apt-X audio
      -- 1.0.62379.2.2.1.9
      aptXAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "APT-X audio."
         ::= { audioSignalFormat 9 }


--  4.1.2.11 enhanced apt-X audio
      -- 1.0.62379.2.2.1.10
      enhAptXAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Enhanced APT-X audio."
         ::= { audioSignalFormat 10 }


--  4.1.2.12 J41 audio
      -- 1.0.62379.2.2.1.11
      j41Audio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J41 audio."
         ::= { audioSignalFormat 11 }


      -- 1.0.62379.2.2.1.11.1
      j41ALawA OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J41 audio using A-law companding, variant A."
         ::= { j41Audio 1 }
```

```
      -- 1.0.62379.2.2.1.11.2
      j41ALawB OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J41 audio using A-law companding, variant B."
         ::= { j41Audio 2 }


      -- 1.0.62379.2.2.1.11.3
      j41Nic OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J41 audio using near instantaneous companding."
         ::= { j41Audio 3 }


--  4.1.2.13 J57 audio
      -- 1.0.62379.2.2.1.12
      j57Audio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J57 audio."
         ::= { audioSignalFormat 12 }


      -- 1.0.62379.2.2.1.12.1
      j57H11 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J57 using an H11 channel."
         ::= { j57Audio 1 }


      -- 1.0.62379.2.2.1.12.2
      j57H12 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "J57 using an H12 channel."
         ::= { j57Audio 2 }


--  4.1.2.14 Invalid audio
      -- 1.0.62379.2.2.1.13
      invalidAudio OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Indicates an error, such as an inability to decode a signal
            earlier in the chain."
         ::= { audioSignalFormat 13 }


--  4.2 Audio transport format definitions
      -- 1.0.62379.2.2.2
      audioTransportFormat OBJECT IDENTIFIER ::= { audioFormat 2 }
```

```
      -- 1.0.62379.2.2.2.0
      unspecifiedTransport OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Unspecified transport."
         ::= { audioTransportFormat 0 }


      -- 1.0.62379.2.2.2.1
      analogue OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Analogue transport."
         ::= { audioTransportFormat 1 }


      -- 1.0.62379.2.2.2.2
      aes3 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "AES3 transport."
         ::= { audioTransportFormat 2 }


      -- 1.0.62379.2.2.2.3
      aes10 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "AES10 transport."
         ::= { audioTransportFormat 3 }


      -- 1.0.62379.2.2.2.4
      aes50 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "AES50 transport."
         ::= { audioTransportFormat 4 }


--  Metadata
      -- 1.0.62379.2.2.3
      audioMetadataFormat OBJECT IDENTIFIER ::= { audioFormat 3 }


      -- 1.0.62379.2.2.3.0
      unspecifiedMetadata OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Unspecified metadata."
         ::= { audioMetadataFormat 0 }



   END
```

Annex F: (Informative) Machine-readable video format definitions

This annex provides a machine-readable version of the video data format definitions specific to this standard which is intended to be interpretable by standard MIB browsing software tools. If there is any inconsistency between this annex and the appropriate clauses in IEC 62379-3 Ed.1 (or later), the appropriate clauses in IEC 62379-3 Ed.1 (or later) take precedence.

The format used to describe the textual conventions definitions conforms to IETF STD 58 (SMIv2).

Note: This annex is not intended to cover every format permitted by the definitions in the appropriate clauses of IEC 62379-3 Ed.1 (or later).

```
IEC62379-3-FORMATS DEFINITIONS ::= BEGIN

   IMPORTS
      iec62379
         FROM IEC62379-1-MIB
      MODULE-IDENTITY, OBJECT-IDENTITY
         FROM SNMPv2-SMI
      TEXTUAL-CONVENTION
         FROM SNMPv2-TC;


   -- 1.0.62379.3.2
   videoFormat MODULE-IDENTITY
      LAST-UPDATED "201106211200Z"    -- June 21, 2011 at 12:00 GMT
      ORGANIZATION
         "IEC PT62379"
      CONTACT-INFO
         "Not specified"
      DESCRIPTION
         "The video format identifiers defined in clause 4 of
         IEC 62379-3 Ed.1.

         Note that although the video formats defined here have been specified
         for use by the EBU ECN-IPM group, they are likely
         to be usable elsewhere. The arrangement also allows the set
         of formats to be easily expanded to include future formats.
         These formats are currently used in IEC 62379-7, Measurements
         for EBU ECN-IPM."
      REVISION "201106101200Z"     -- June 10, 2011 at 12:00 GMT
      DESCRIPTION
         "Moved invalidVideo up and added additional coding types VP8 and
         H264 Scalable Extn and also Aspect Ratio entries.
         Removed Video bit rate types and value."
      ::= { video 2 }



--
-- Textual conventions
--
```

```
-- 4.1 Video signal format definitions
-- 4.1.1 Video parameters
      FrameRate ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "The video frame rate in Hz."
         SYNTAX INTEGER (0..2147483647)

      SourceType ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An enumeration describing the video definition source type."
         SYNTAX INTEGER
            {
            unspecified(0),
            sd(1),
            hd(2)
            }
-- {
-- unspecified(0),
-- sd(1),
-- hd(2)
-- } (unspecified..hd)
--
      LineResolution ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An integer representing the number of vertical lines of resolution."
         SYNTAX INTEGER (0..2147483647)

      ScanType ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An enumeration describing the video scan type.
            psf = Progressive Segmented Frame"
         SYNTAX INTEGER
            {
            unspecified(0),
            progressive(1),
            interlaced(2),
            psf(3)
            }
-- {
-- unspecified(0),
-- progressive(1),
-- interlaced(2),
-- psf(3)
-- } (unspecified..psf)
--
      CodingType ::= TEXTUAL-CONVENTION
         STATUS current
         DESCRIPTION
            "An enumeration describing the video coding type."
         SYNTAX INTEGER
            {
            unspecified(0),
            uncompressed(1),
```

```
          mpeg2(2),
          h264(3),
          jpeg2000(4),
          smptecvc2(5),
          vP8(6),
          h264ScaleExtn(7)
          }
-- {
-- unspecified(0),
-- uncompressed(1),
-- mpeg2(2),
-- h264(3),
-- jpeg2000(4),
-- smptecvc2(5),
-- vP8(6),
-- h264ScaleExtn(7)
-- } (unspecified..h264ScaleExtn)
--
     SourceAspectRatio ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An enumeration describing the source aspect ratio.

           fourByThree = 4:3
           sixteenByNine = 16:9
           twoPointTwoOne = 2.12:1"
        SYNTAX INTEGER
           {
           unspecified(0),
           fourByThree(43),
           sixteenByNine(169),
           twoPointTwoOne(221)
           }
-- {
-- unspecified(0),
-- fourByThree(43),
-- sixteenByNine(169),
-- twoPointTwoOne(221)
-- } (unspecified..twoPointTwoOne)
--
     ActiveFormatDescriptionCodes ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
           "An integer representing the active format description codes for video used
           with the range of source aspect ratios."
        SYNTAX INTEGER (0..15)


--
-- Node definitions
--

     -- 1.0.62379.3
     video OBJECT IDENTIFIER ::= { iec62379 3 }


-- 4.1.2 Video signal formats
```

```
      -- 1.0.62379.3.2.1
      videoSignalFormat OBJECT IDENTIFIER ::= { videoFormat 1 }


-- 4.1.2.1 Unspecified video
      -- 1.0.62379.3.2.1.0
      unspecifiedVideo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "VideoFormat wildcard value - any supported format allowed."
         ::= { videoSignalFormat 0 }


-- 4.1.2.2 No video
      -- 1.0.62379.3.2.1.1
      noVideo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Indicates the output is non-existant."
         ::= { videoSignalFormat 1 }


-- 4.1.2.3 Invalid video
      -- 1.0.62379.3.2.1.2
      invalidVideo OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Indicates an error, such as an inability to decode a signal
            earlier in the chain."
         ::= { videoSignalFormat 2 }


-- 4.1.2.4 Video Source
      -- 1.0.62379.3.2.1.3
      videoSource OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Video source descriptions."
         ::= { videoSignalFormat 3 }


-- 4.1.1.2 Frame Rate
      -- 1.0.62379.3.2.1.3.23
      frameRate23 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Video at a Frame Rate of 23Hz.
            Actual value is 23.976Hz.

            Reserved for future use."
         ::= { videoSource 23 }


-- 4.1.1.2 Frame Rate
      -- 1.0.62379.3.2.1.3.24
      frameRate24 OBJECT-IDENTITY
         STATUS current
```

```
        DESCRIPTION
           "Video at a Frame Rate of 24Hz.

           Reserved for future use."
        ::= { videoSource 24 }


-- 4.1.1.2 Frame Rate
      -- 1.0.62379.3.2.1.3.25
      frameRate25 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video at a Frame Rate of 25Hz."
        ::= { videoSource 25 }


-- 4.1.1.3 Source Type
      -- 1.0.62379.3.2.1.3.25.1
      frameRate25SD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 25Hz."
        ::= { frameRate25 1 }


-- 4.1.1.4 Line resolution
      -- 1.0.62379.3.2.1.3.25.1.625
      frameRate25SDat625 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 25Hz at 625 vertical lines of
           resolution."
        ::= { frameRate25SD 625 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.1.625.1
      frameRate25SDat625P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 25Hz at 625 vertical lines of
           resolution with progressive scanning."
        ::= { frameRate25SDat625 1 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.1.625.2
      frameRate25SDat625I OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 25Hz at 625 vertical lines of
           resolution with interlaced scanning."
        ::= { frameRate25SDat625 2 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.1.625.3
```

```
      frameRate25SDat625S OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Standard Definition Video at a Frame Rate of 25Hz at 625 vertical lines of
            resolution with progressive segmented frame."
         ::= { frameRate25SDat625 3 }


-- 4.1.1.3 Source Type
      -- 1.0.62379.3.2.1.3.25.2
      frameRate25HD OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "High Definition Video at a Frame Rate of 25Hz."
         ::= { frameRate25 2 }


-- 4.1.1.4 Line resolution
      -- 1.0.62379.3.2.1.3.25.2.1080
      frameRate25HDat1080 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "High Definition Video at a Frame Rate of 25Hz at 1080 vertical lines of
            resolution."
         ::= { frameRate25HD 1080 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.2.1080.1
      frameRate25HDat1080P OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "High Definition Video at a Frame Rate of 25Hz at 1080 vertical lines of
            resolution with progressive scanning."
         ::= { frameRate25HDat1080 1 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.2.1080.2
      frameRate25HDat1080I OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "High Definition Video at a Frame Rate of 25Hz at 1080 vertical lines of
            resolution with interlaced scanning."
         ::= { frameRate25HDat1080 2 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.25.2.1080.3
      frameRate25HDat1080S OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "High Definition Video at a Frame Rate of 25Hz at 1080 vertical lines of
            resolution with progressive segmented frame."
         ::= { frameRate25HDat1080 3 }
```

```
-- 4.1.1.2 Frame Rate
     -- 1.0.62379.3.2.1.3.29
     frameRate29 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video at a Frame Rate of 29Hz.
           Actual value is 29.97Hz."
        ::= { videoSource 29 }


-- 4.1.1.3 Source Type
     -- 1.0.62379.3.2.1.3.29.1
     frameRate29SD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 29Hz."
        ::= { frameRate29 1 }


-- 4.1.1.4 Line resolution
     -- 1.0.62379.3.2.1.3.29.1.525
     frameRate29SDat525 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 29Hz at 525 vertical lines of
           resolution."
        ::= { frameRate29SD 525 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.1.525.1
     frameRate29SDat525P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 29Hz at 525 vertical lines of
           resolution with progressive scanning."
        ::= { frameRate29SDat525 1 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.1.525.2
     frameRate29SDat525I OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 29Hz at 525 vertical lines of
           resolution with interlaced scanning."
        ::= { frameRate29SDat525 2 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.1.525.3
     frameRate29SDat525S OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Standard Definition Video at a Frame Rate of 29Hz at 525 vertical lines of
           resolution with progressive segmented frame."
        ::= { frameRate29SDat525 3 }
```

```
-- 4.1.1.3 Source Type
     -- 1.0.62379.3.2.1.3.29.2
     frameRate29HD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 29Hz."
        ::= { frameRate29 2 }


-- 4.1.1.4 Line resolution
     -- 1.0.62379.3.2.1.3.29.2.1080
     frameRate29HDat1080 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 29Hz at 1080 vertical lines of
           resolution."
        ::= { frameRate29HD 1080 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.2.1080.1
     frameRate29HDat1080P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 29Hz at 1080 vertical lines of
           resolution with progressive scanning."
        ::= { frameRate29HDat1080 1 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.2.1080.2
     frameRate29HDat1080I OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 29Hz at 1080 vertical lines of
           resolution with interlaced scanning."
        ::= { frameRate29HDat1080 2 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.29.2.1080.3
     frameRate29HDat1080S OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 29Hz at 1080 vertical lines of
           resolution with progressive segmented frame."
        ::= { frameRate29HDat1080 3 }


-- 4.1.1.2 Frame Rate
     -- 1.0.62379.3.2.1.3.30
     frameRate30 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video at a Frame Rate of 30Hz.
```

```
          Reserved for future use."
        ::= { videoSource 30 }


-- 4.1.1.2 Frame Rate
     -- 1.0.62379.3.2.1.3.50
     frameRate50 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video at a Frame Rate of 50Hz."
        ::= { videoSource 50 }


-- 4.1.1.3 Source Type
     -- 1.0.62379.3.2.1.3.50.2
     frameRate50HD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 50Hz."
        ::= { frameRate50 2 }


-- 4.1.1.4 Line resolution
     -- 1.0.62379.3.2.1.3.50.2.720
     frameRate50HDat720 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 50Hz at 720 vertical lines of
           resolution."
        ::= { frameRate50HD 720 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.50.2.720.1
     frameRate50HDat720P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 50Hz at 720 vertical lines of
           resolution with progressive scanning."
        ::= { frameRate50HDat720 1 }


-- 4.1.1.4 Line resolution
     -- 1.0.62379.3.2.1.3.50.2.1080
     frameRate50HDat1080 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 50Hz at 1080 vertical lines of
           resolution."
        ::= { frameRate50HD 1080 }


-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.50.2.1080.1
     frameRate50HDat1080P OBJECT-IDENTITY
        STATUS current
```

```
        DESCRIPTION
          "High Definition Video at a Frame Rate of 50Hz at 1080 vertical lines of
          resolution with progressive scanning."
        ::= { frameRate50HDat1080 1 }


-- 4.1.1.2 Frame Rate
      -- 1.0.62379.3.2.1.3.59
      frameRate59 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "Video at a Frame Rate of 59Hz.
          Actual value is 59.94Hz.

          Colloquially also known as 60!"
        ::= { videoSource 59 }


-- 4.1.1.3 Source Type
      -- 1.0.62379.3.2.1.3.59.2
      frameRate59HD OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "High Definition Video at a Frame Rate of 59Hz."
        ::= { frameRate59 2 }


-- 4.1.1.4 Line resolution
      -- 1.0.62379.3.2.1.3.59.2.720
      frameRate59HDat720 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "High Definition Video at a Frame Rate of 59Hz at 720 vertical lines of
          resolution."
        ::= { frameRate59HD 720 }


-- 4.1.1.5 Scan Type
      -- 1.0.62379.3.2.1.3.59.2.720.1
      frameRate59HDat720P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "High Definition Video at a Frame Rate of 59Hz at 720 vertical lines of
          resolution with progressive scanning."
        ::= { frameRate59HDat720 1 }


-- 4.1.1.4 Line resolution
      -- 1.0.62379.3.2.1.3.59.2.1080
      frameRate59HDat1080 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
          "High Definition Video at a Frame Rate of 59Hz at 1080 vertical lines of
          resolution."
        ::= { frameRate59HD 1080 }
```

```
-- 4.1.1.5 Scan Type
     -- 1.0.62379.3.2.1.3.59.2.1080.1
     frameRate59HDat1080P OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "High Definition Video at a Frame Rate of 59Hz at 1080 vertical lines of
           resolution with progressive scanning."
        ::= { frameRate59HDat1080 1 }


-- 4.1.1.2 Frame Rate
     -- 1.0.62379.3.2.1.3.60
     frameRate60 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video at a Frame Rate 60Hz.

           Reserved for future use."
        ::= { videoSource 60 }


-- 4.1.2.5 Video Coding Type
     -- 1.0.62379.3.2.1.4
     videoCodingType OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video coding type (if applicable)."
        ::= { videoSignalFormat 4 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.0
     codingTypeUnspecified OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Unspecified video."
        ::= { videoCodingType 0 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.1
     codingTypeUncompressed OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Uncompressed video."
        ::= { videoCodingType 1 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.2
     codingTypeMPEG2 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "MPEG2 coded video."
        ::= { videoCodingType 2 }
```

```
-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.3
     codingTypeH264 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "H264 coded video."
        ::= { videoCodingType 3 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.4
     codingTypeJPEG2000 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "JPEG2000 coded video."
        ::= { videoCodingType 4 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.5
     codingTypeSMPTEVC2 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "SMPTE-VC2 coded video (DIRAC)"
        ::= { videoCodingType 5 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.6
     codingTypeVP8 OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "VP8 coded video"
        ::= { videoCodingType 6 }


-- 4.1.1.6 Coding Type
     -- 1.0.62379.3.2.1.4.7
     codingTypeH264ScaleExtn OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "H.264 Advanced Video Coding (AVC) with
           Scalable Video Coding (SVC) coded video"
        ::= { videoCodingType 7 }


-- 4.1.2.6 Video Aspect Ratio
     -- 1.0.62379.3.2.1.5
     aspectRatio OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
           "Video aspect ratio descriptions."
        ::= { videoSignalFormat 5 }


-- 4.1.1.7 Source Aspect Ratio
     -- 1.0.62379.3.2.1.5.0
```

```
      sourceAspectRatioUnspecified OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "The video source aspect ratio is unspecified."
         ::= { aspectRatio 0 }


-- 4.1.1.8 Active Format Description
      -- 1.0.62379.3.2.1.5.0.0
      srceARUnspecifiedAFD0000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatioUnspecified 0 }


-- 4.1.1.7 Source Aspect Ratio
      -- 1.0.62379.3.2.1.5.43
      sourceAspectRatio4x3 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Video with an aspect ratio of 4:3."
         ::= { aspectRatio 43 }


-- 4.1.1.8 Active Format Description
      -- 1.0.62379.3.2.1.5.43.0
      srceAR4x3AFD0000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 0 }


      -- 1.0.62379.3.2.1.5.43.1
      srceAR4x3AFD0001 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 1 }


      -- 1.0.62379.3.2.1.5.43.2
      srceAR4x3AFD0010 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 2 }


      -- 1.0.62379.3.2.1.5.43.3
      srceAR4x3AFD0011 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 3 }
```

```
-- 1.0.62379.3.2.1.5.43.4
srceAR4x3AFD0100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 4 }


-- 1.0.62379.3.2.1.5.43.5
srceAR4x3AFD0101 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 5 }


-- 1.0.62379.3.2.1.5.43.6
srceAR4x3AFD0110 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 6 }


-- 1.0.62379.3.2.1.5.43.7
srceAR4x3AFD0111 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 7 }


-- 1.0.62379.3.2.1.5.43.8
srceAR4x3AFD1000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 8 }


-- 1.0.62379.3.2.1.5.43.9
srceAR4x3AFD1001 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 9 }


-- 1.0.62379.3.2.1.5.43.10
srceAR4x3AFD1010 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio4x3 10 }
```

```
      -- 1.0.62379.3.2.1.5.43.11
      srceAR4x3AFD1011 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 11 }


      -- 1.0.62379.3.2.1.5.43.12
      srceAR4x3AFD1100 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 12 }


      -- 1.0.62379.3.2.1.5.43.13
      srceAR4x3AFD1101 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 13 }


      -- 1.0.62379.3.2.1.5.43.14
      srceAR4x3AFD1110 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 14 }


      -- 1.0.62379.3.2.1.5.43.15
      srceAR4x3AFD1111 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio4x3 15 }


-- 4.1.1.7 Source Aspect Ratio
      -- 1.0.62379.3.2.1.5.169
      sourceAspectRatio16x9 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Video with an aspect ratio of 16:9."
         ::= { aspectRatio 169 }


-- 4.1.1.8 Active Format Description
      -- 1.0.62379.3.2.1.5.169.0
      srceAR16x9AFD0000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio16x9 0 }
```

```
-- 1.0.62379.3.2.1.5.169.1
srceAR16x9AFD0001 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 1 }


-- 1.0.62379.3.2.1.5.169.2
srceAR16x9AFD0010 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 2 }


-- 1.0.62379.3.2.1.5.169.3
srceAR16x9AFD0011 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 3 }


-- 1.0.62379.3.2.1.5.169.4
srceAR16x9AFD0100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 4 }


-- 1.0.62379.3.2.1.5.169.5
srceAR16x9AFD0101 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 5 }


-- 1.0.62379.3.2.1.5.169.6
srceAR16x9AFD0110 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 6 }


-- 1.0.62379.3.2.1.5.169.7
srceAR16x9AFD0111 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 7 }
```

```
-- 1.0.62379.3.2.1.5.169.8
srceAR16x9AFD1000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 8 }


-- 1.0.62379.3.2.1.5.169.9
srceAR16x9AFD1001 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 9 }


-- 1.0.62379.3.2.1.5.169.10
srceAR16x9AFD1010 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 10 }


-- 1.0.62379.3.2.1.5.169.11
srceAR16x9AFD1011 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 11 }


-- 1.0.62379.3.2.1.5.169.12
srceAR16x9AFD1100 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 12 }


-- 1.0.62379.3.2.1.5.169.13
srceAR16x9AFD1101 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 13 }


-- 1.0.62379.3.2.1.5.169.14
srceAR16x9AFD1110 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio16x9 14 }


-- 1.0.62379.3.2.1.5.169.15
```

```
      srceAR16x9AFD1111 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio16x9 15 }


-- 4.1.1.7 Source Aspect Ratio
      -- 1.0.62379.3.2.1.5.221
      sourceAspectRatio2pt21 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "Video with an aspect ratio of 2.21:1"
         ::= { aspectRatio 221 }


-- 4.1.1.8 Active Format Description
      -- 1.0.62379.3.2.1.5.221.0
      srceAR2pt21AFD0000 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio2pt21 0 }


      -- 1.0.62379.3.2.1.5.221.1
      srceAR2pt21AFD0001 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio2pt21 1 }


      -- 1.0.62379.3.2.1.5.221.2
      srceAR2pt21AFD0010 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio2pt21 2 }


      -- 1.0.62379.3.2.1.5.221.3
      srceAR2pt21AFD0011 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio2pt21 3 }


      -- 1.0.62379.3.2.1.5.221.4
      srceAR2pt21AFD0100 OBJECT-IDENTITY
         STATUS current
         DESCRIPTION
            "See SMPTE ST 2016-1:2009 for code descriptions."
         ::= { sourceAspectRatio2pt21 4 }
```

```
-- 1.0.62379.3.2.1.5.221.5
srceAR2pt21AFD0101 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 5 }


-- 1.0.62379.3.2.1.5.221.6
srceAR2pt21AFD0110 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 6 }


-- 1.0.62379.3.2.1.5.221.7
srceAR2pt21AFD0111 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 7 }


-- 1.0.62379.3.2.1.5.221.8
srceAR2pt21AFD1000 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 8 }


-- 1.0.62379.3.2.1.5.221.9
srceAR2pt21AFD1001 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 9 }


-- 1.0.62379.3.2.1.5.221.10
srceAR2pt21AFD1010 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 10 }


-- 1.0.62379.3.2.1.5.221.11
srceAR2pt21AFD1011 OBJECT-IDENTITY
   STATUS current
   DESCRIPTION
      "See SMPTE ST 2016-1:2009 for code descriptions."
   ::= { sourceAspectRatio2pt21 11 }


-- 1.0.62379.3.2.1.5.221.12
```

```
   srceAR2pt21AFD1100 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "See SMPTE ST 2016-1:2009 for code descriptions."
      ::= { sourceAspectRatio2pt21 12 }


   -- 1.0.62379.3.2.1.5.221.13
   srceAR2pt21AFD1101 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "See SMPTE ST 2016-1:2009 for code descriptions."
      ::= { sourceAspectRatio2pt21 13 }


   -- 1.0.62379.3.2.1.5.221.14
   srceAR2pt21AFD1110 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "See SMPTE ST 2016-1:2009 for code descriptions."
      ::= { sourceAspectRatio2pt21 14 }


   -- 1.0.62379.3.2.1.5.221.15
   srceAR2pt21AFD1111 OBJECT-IDENTITY
      STATUS current
      DESCRIPTION
         "See SMPTE ST 2016-1:2009 for code descriptions."
      ::= { sourceAspectRatio2pt21 15 }

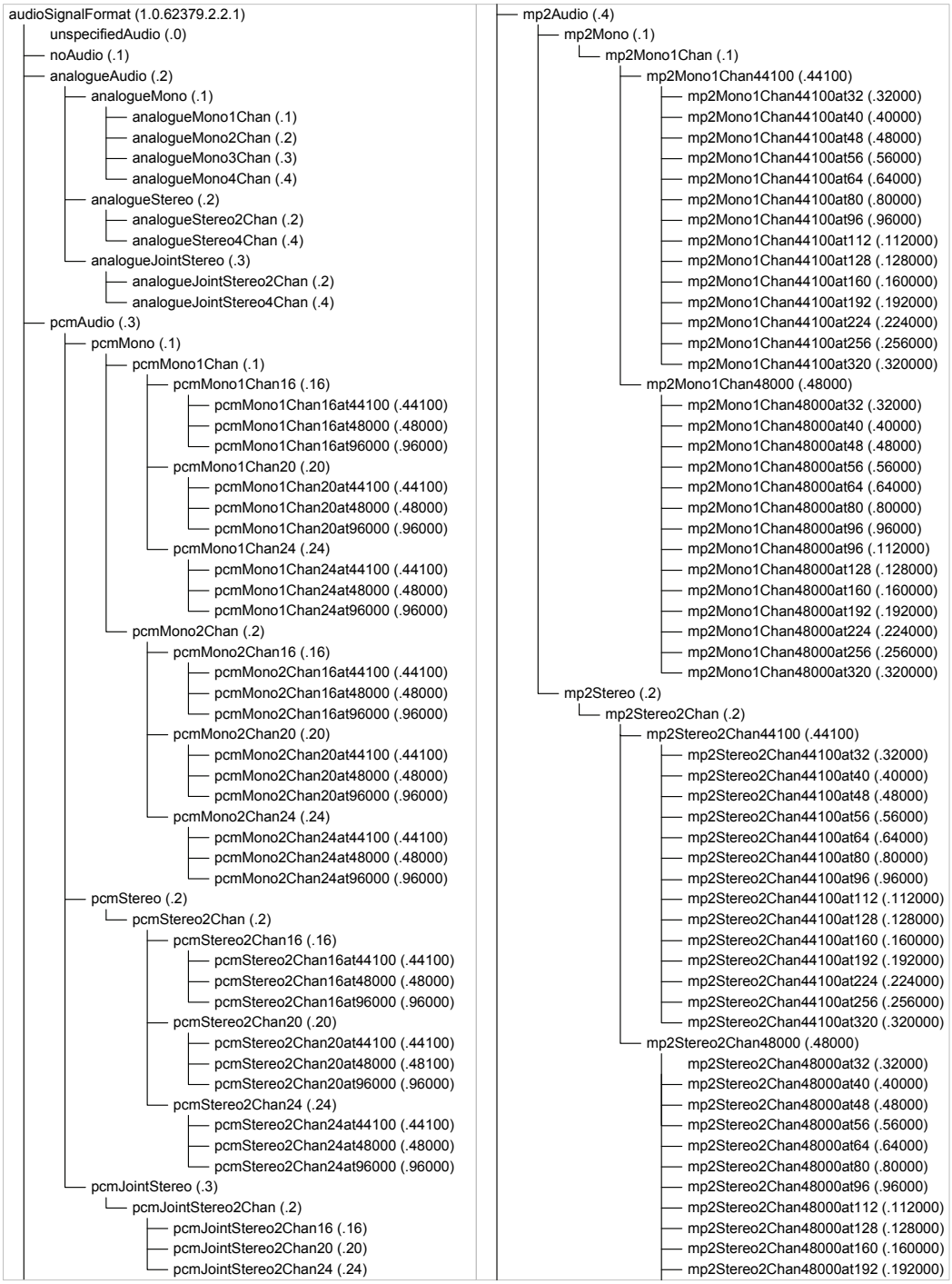


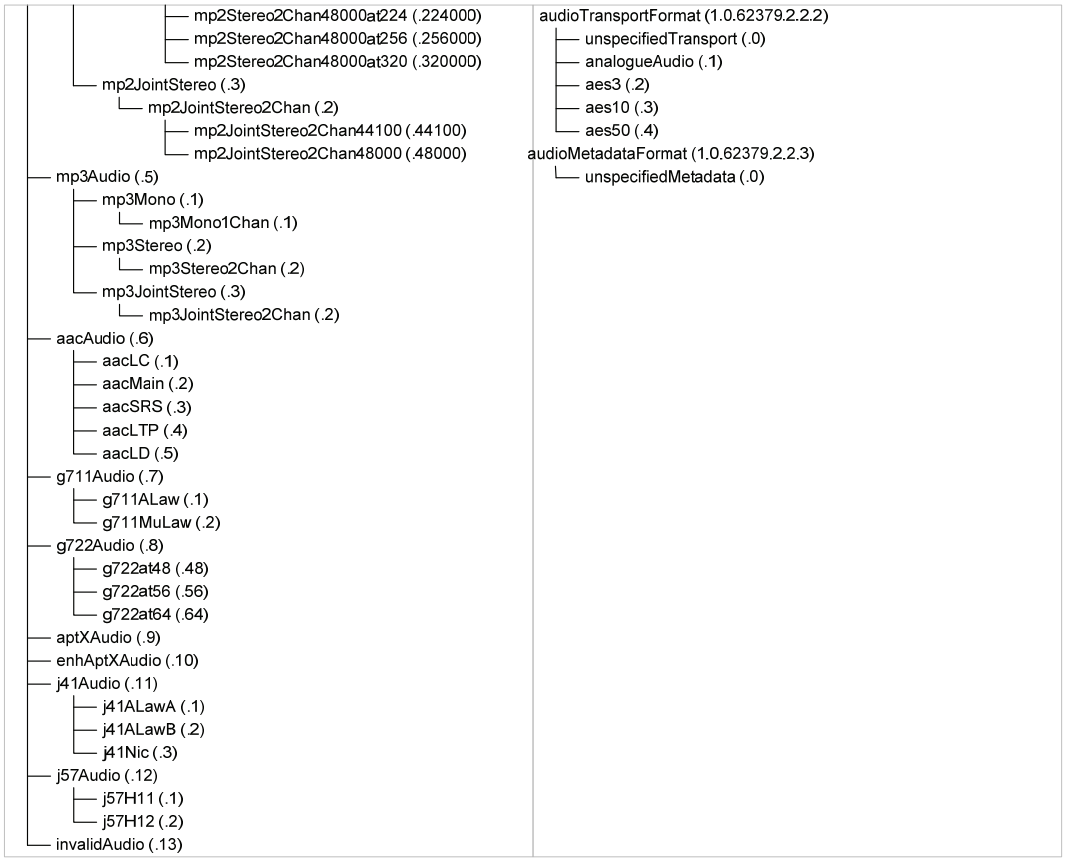
END
```

Annex G: (Informative) Tree of example audio formats

This annex shows a graphical tree of the example audio formats listed in Annex E. If there is any inconsistency between this annex and clause 4 of IEC 62379-2, clause 4 of IEC 62379-2 takes precedence.

Note: This annex is not intended to cover every format permitted by the definitions in clause 4 of IEC 62379-2.




mp2Stereo2Chan48000at224 (.224000)
mp2Stereo2Chan48000at256 (.256000)
mp2Stereo2Chan48000at320 (.320000)
mp2JointStereo (.3)
mp2JointStereo2Chan (.2)
mp2JointStereo2Chan44100 (.44100)
mp2JointStereo2Chan48000 (.48000)
mp3Audio (.5)
mp3Mono (.1)
mp3Mono1Chan (.1)
mp3Stereo (.2)
mp3Stereo2Chan (.2)
mp3JointStereo (.3)
mp3JointStereo2Chan (.2)
aacAudio (.6)
aacLC (.1)
aacMain (.2)
aacSRS (.3)
aacLTP (.4)
aacLD (.5)
g711Audio (.7)
g711ALaw (.1)
g711MuLaw (.2)
g722Audio (.8)
g722at48 (.48)
g722at56 (.56)
g722at64 (.64)
aptXAudio (.9)
enhAptXAudio (.10)
j41Audio (.11)
j41ALawA (.1)
j41ALawB (.2)
j41Nic (.3)
j57Audio (.12)
j57H11 (.1)
j57H12 (.2)
invalidAudio (.13)
audioTransportFormat (1.0.62379.2.2.2)
unspecifiedTransport (.0)
analogueAudio (.1)
aes3 (.2)
aes10 (.3)
aes50 (.4)
audioMetadataFormat (1.0.62379.2.2.3)
unspecifiedMetadata (.0)

Annex H: (Informative) Tree of example video formats

This annex shows a graphical tree of the example video formats listed in Annex F. If there is any inconsistency between this annex and the appropriate clauses in IEC 62379-3 Ed.1 (or later), the appropriate clauses in IEC 62379-3 Ed.1 (or later) take precedence.

Note: This annex is not intended to cover every format permitted by the definitions in the appropriate clause of IEC 62379-3 Ed.1 (or later).

