

INTERNATIONAL STANDARD



Common control interface for networked digital audio and video products – Part 7: Measurements



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2015 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in 15 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

More than 60 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.



IEC 62379-7

Edition 1.0 2015-06

INTERNATIONAL STANDARD



Common control interface for networked digital audio and video products – Part 7: Measurements

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 33.160; 35.100

ISBN 978-2-8322-2753-4

Warning! Make sure that you obtained this publication from an authorized distributor.

CONTENTS

FOREWORD.....	5
INTRODUCTION.....	7
1 Scope.....	10
2 Normative references.....	10
3 Terms, definitions and abbreviations	10
3.1 Terms and definitions	10
3.2 Abbreviations	10
4 Audio format definitions	11
5 Video format definitions	11
6 MIB definitions for measurement information blocks	11
6.1 General.....	11
6.2 Type definitions.....	12
6.2.1 General	12
6.2.2 Textual conventions	12
6.2.3 Sequences.....	13
6.3 Network measurement information blocks	14
6.3.1 Network measurement information block structure	14
6.3.2 nMtBlockTable	15
6.3.3 nMtBlockEntry.....	15
6.3.4 nMtBlockId.....	15
6.3.5 nMtIfIndex.....	15
6.3.6 nMtTxRxPoint	15
6.3.7 nMtNetworkType	16
6.3.8 nMtTransportType	16
6.3.9 nMtTxRxAddr	16
6.3.10 nMtPortNumber.....	16
6.3.11 nMtIGMPVersion.....	16
6.3.12 nMtSIPServerAddr	16
6.4 Audio measurement information blocks.....	17
6.4.1 Audio measurement information block structure	17
6.4.2 aMtBlockTable	17
6.4.3 aMtBlockEntry.....	17
6.4.4 aMtBlockId.....	18
6.4.5 aMtAudioComponentNumber	18
6.4.6 aMtNetworkBlockId	18
6.4.7 aMtAudioStatus.....	18
6.4.8 aMtAudioSignalFormat	18
6.4.9 aMtAudioPIId	18
6.4.10 aMtIfIndex.....	18
6.4.11 aMtFECType.....	19
6.4.12 aMtFECLengthDimension	19
6.5 Video measurement information blocks.....	19
6.5.1 Video measurement information block structure	19
6.5.2 vMtBlockTable	20

6.5.3	vMtBlockEntry	20
6.5.4	vMtBlockId	20
6.5.5	vMtAudioBlockId	20
6.5.6	vMtNetworkBlockId	20
6.5.7	vMtVideoStatus	20
6.5.8	vMtVideoSourceFormat	20
6.5.9	vMtVideoCodingType	21
6.5.10	vMtVideoBitRateType	21
6.5.11	vMtVideoBitRate	21
6.5.12	vMtVideoAspectRatio	21
6.5.13	vMtFECType	21
6.5.14	vMtFECLengthDimension	21
6.5.15	vMtTrickModeSupport	21
6.6	Receiver point measurement information block	21
6.6.1	Receiver measurement information block structure	21
6.6.2	rxPointTable	22
6.6.3	rxPointEntry	23
6.6.4	rxPointBlockId	23
6.6.5	rxPointNetworkBlockId	23
6.6.6	rxPointBufferSize	23
6.6.7	rxPointBufferOccupancyTime	23
6.6.8	rxPointBufferOccupancyPcnt	23
6.6.9	rxPointMDI	23
6.6.10	rxPointTSDF	23
6.7	Temperature measurement information block	23
6.7.1	Temperature measurement information block structure	23
6.7.2	temperatureTable	24
6.7.3	temperatureEntry	24
6.7.4	temperatureBlockId	24
6.7.5	temperatureLocnNumber	25
6.7.6	temperatureLocation	25
6.7.7	temperatureTrend	25
6.7.8	temperatureStatus	25
6.7.9	temperatureLowWarning	25
6.7.10	temperatureHighWarning	25
6.7.11	temperatureLowCritical	25
6.7.12	temperatureHighCritical	25
Annex A (informative)	Machine-readable measurement block definitions	26
Annex B (informative)	Machine-readable textual conventions definitions	44
Annex C (informative)	Worked example	48
C.1	Overview	48
C.2	Example 1	48
C.2.1	General	48
C.2.2	Block table	48
C.2.3	Mixer block	50
C.2.4	Multiple functionality device	51
C.2.5	Summary of tables	54
Bibliography	55

Figure 1 – Relationships between ECN groups ACIP, VCIP and IPM	8
Figure 2 – Network measurement information block	14
Figure 3 – Audio measurement information block.....	17
Figure 4 – Video measurement information block.....	19
Figure 5 – Receiver measurement information block	22
Figure 6 – Temperature measurement information block	24
Figure C.1 – Example of a modified audio device.....	48
Figure C.2 – Annotated connector diagram.....	49
Figure C.3 – Mixer section	50
Figure C.4 – Single device with multiple functionality	52
Figure C.5 – Measurement blockIds and their associated media components	52
Figure C.6 – Single device with multiple functionality	53
Table 1 – Managed objects for network measurement information blocks	15
Table 2 – Managed objects for audio measurement information blocks.....	17
Table 3 – Managed objects for video measurement information blocks.....	20
Table 4 – Managed objects for receiver measurement information blocks.....	22
Table 5 – Managed objects conveying temperature information about the unit.....	24
Table C.1 – Main block Id table	49
Table C.2 – Mixer related block Id table	50
Table C.3 – Mixer block tables	51
Table C.4 – Addition of measurement block Ids	52
Table C.5 – Video measurement table.....	53
Table C.6 – Network measurement table	53
Table C.7 – Audio measurement table	53
Table C.8 – Table summary	54

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**COMMON CONTROL INTERFACE FOR NETWORKED
DIGITAL AUDIO AND VIDEO PRODUCTS –****Part 7: Measurements****FOREWORD**

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 62379-7 has been prepared by technical area 4: Digital system interfaces and protocols of IEC technical committee 100: Audio, video and multimedia systems and equipment.

The text of this standard is based on the following documents:

CDV	Report on voting
100/2168/CDV	100/2338/RVC

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts in the IEC 62379 series, published under the general title *Common control interface for networked digital audio and video products*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

A bilingual version of this publication may be issued at a later date.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

INTRODUCTION

IEC 62379 specifies the common control interface, a protocol for managing equipment which conveys audio and/or video across digital networks.

An introduction to the common control interface is given in IEC 62739-1.

This part of IEC 62379 specifies those aspects that are specific for using the block structure as defined in IEC 62379-1, for standardising the collection method of audio and video parameters for use by the European Broadcasting Union Expert Communities Networks – Internet Protocol (IP) Measurements (EBU ECN-IPM) Group.

The collection of network related parameters may be outside the scope of this standard. These are expected to be collected from the standard Internet Engineering Task Force (IETF) Management Information Base (MIBs) that are generally present in most (if not all) networked equipment. Some specific network parameters are included that are not obtainable from existing standard IETF MIBs.

Structure of the family of standards

IEC 62379 specifies the common control interface, a protocol for managing networked audiovisual equipment. It is intended to include the following Parts:

Part 1: General

Part 2: Audio

Part 3: Video

Part 4: Data

Part 5: Transmission over networks

Part 6: Packet transfer service

Part 7: Measurement

Part 1 specifies aspects which are common to all equipment.

Parts 2 to 4 specify control of internal functions specific to equipment carrying particular types of live media. Part 4 does not refer to packet data such as the control messages themselves.

Part 5 specifies control of transmission of these media over each individual network technology. It includes network specific management interfaces along with network specific control elements that integrate into the control framework.

Part 6 specifies carriage of control and status messages and non-audiovisual data over transports that do not support audio and video, such as RS232 serial links, with (as with Part 5) a separate subpart for each technology.

Part 7 specifies those aspects that are specific to the measurement requirements of the EBU ECN-IPM Group.

An introduction to the common control interface is given in IEC 62739-1.

Description, aims and requirements of the EBU ECN-IPM Group

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, EBU set up two groups, *Expert Communities Networks Audio contribution over IP (ECN-ACIP)* and *Expert Communities Networks – Video contribution over IP (ECN-VCIP)*, with the tasks of drawing up recommended codes of practice¹.

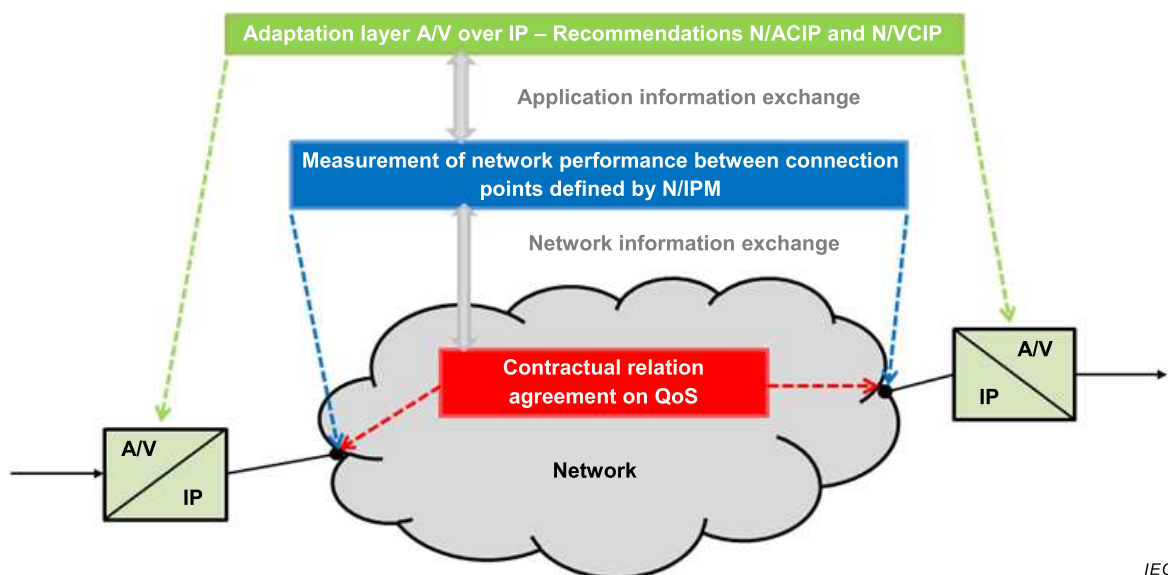


Figure 1 – Relationships between ECN groups ACIP, VCIP and IPM

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 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 Figure 1.

The goals of ECN-IPM Group were to

- define a quality of service classification to achieve requested A/V transmission quality for broadcast applications,
- standardise network information exchange between EBU members and Telecom suppliers,
- 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.

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

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

Two EBU technical publications have been produced by the ECN-IPM Group.

The parameters and new MIB information may be found in EBU-Tech 3345, End-to-End IP Network Measurement for Broadcast Applications – Parameters & Management Information Base (MIB), Geneva, July 2011.

A description of the software mechanism, EisStream², may be found in EBU-Tech 3346, End-to-End IP Network Measurement for Broadcast Applications – EisStream Software package description, Geneva, July 2011. The software is written in Java and it provides physical path tracing for IP traffic using SNMP.

This part of IEC 62379 and other related parts of IEC 62379, constitute the standards upon which Section 3 of EBU-Tech 3345 is based.

If there is any inconsistency between this standard and Section 3 of EBU-Tech 3345, then IEC 62379-7 and other related parts of IEC 62379, take precedence.

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

COMMON CONTROL INTERFACE FOR NETWORKED DIGITAL AUDIO AND VIDEO PRODUCTS –

Part 7: Measurements

1 Scope

This part of IEC 62379 specifies aspects of the common control interface of IEC 62379-1 that are specific to the measurement of the service experienced by audio and video streams and in particular to the requirements of EBU ECN-IPM Measurements Group.

2 Normative references

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

IEC 62379-1, *Common control interface for networked digital audio and video products – Part 1: General*

IEC 62379-2:2008, *Common control interface for networked digital audio and video products – Part 2: Audio*

IEC 62379-3, *Common control interface for networked audio and video products – Part 3: Video*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62379-1 and IEC 62379-2 apply.

3.2 Abbreviations

ASI	Asynchronous Serial Interface
DF	Delay Factor
FEC	Forward Error Correction
HD	High Definition
IGMP	Internet Group Management Protocol
MDI	Media Delivery Index
MIB	Management Information Base
MLR	Media Loss Rate
OID	Object IDentifier
PID	Programme ID
RTP	Real-Time Protocol
SD	Standard Definition
SIP	Session Initiation Protocol
SNMP	Simple Network Management Protocol
TS-DF	Time Stamped Delay Factor

4 Audio format definitions

At any point in the audio signal chain, the audio data will be in a particular format. For management purposes, the format shall be identified by an object identifier, either a “Common control interface standard” object identifier as defined and specified in IEC 62379-2 or an object identifier defined elsewhere.

NOTE 1 Permitting audio format identifiers to be defined outside this standard allows use of proprietary formats within the standard protocol and also allows industry standard formats to emerge that may eventually be incorporated into future revisions of this standard.

NOTE 2 The audio signal format definitions specified in IEC 62379-2 are used in a common manner by both audio only units and the one or more audio components associated with a video flow.

5 Video format definitions

At any point in the video signal chain, the video data will be in a particular format. For management purposes, the format should be identified by an object identifier, either a “Common control interface standard” object identifier as defined and specified in IEC 62379-3 or an object identifier defined elsewhere.

NOTE Permitting video format identifiers to be defined outside this standard allows use of proprietary formats within the standard protocol and also allows industry standard formats to emerge that may eventually be incorporated into future revisions of this standard.

6 MIB definitions for measurement information blocks

6.1 General

This clause defines a set of managed object types for representing measurement information functions in network controlled audio/video equipment for the purposes of standardising the collection method of audio, video and some network parameters for use by the EBU ECN-IPM Group.

The format of the definitions is as specified in IEC 62379-1.

For measurement purposes, a piece of audio/video equipment shall be modelled as a number of discrete measurement blocks, as specified in IEC 62379-1. Each measurement block shall have zero or more inputs and zero or more outputs.

NOTE 1 Information is transferred into these objects internally within the equipment utilising this standard, from existing MIB objects, or elsewhere, within the equipment. A management station can access the measurement information using the standard object identifier (OIDs) defined in this standard independent of the manufacturer.

Each measurement block shall be modelled either by one of the standard measurement block types defined in this standard or by a measurement block type defined elsewhere. Associated with each defined block type shall be a (possibly empty) group of managed object types that represent the control functions for that block. A block type shall be identified by the node in the object identifier tree that is the root node for the group of managed object types associated with that block type.

NOTE 2 Permitting measurement block types to be defined outside this standard allows control of proprietary functions using the standard protocol and also allows industry standard block types to emerge that may eventually be incorporated into future revisions of this standard.

NOTE 3 An empty group of managed object types is permitted to allow for blocks that have no associated control functions.

NOTE 4 Annex C contains a worked example of the use of the measurement block structure.

6.2 Type definitions

6.2.1 General

In addition to the types defined in IEC 62379-1, the following types are used to specify the syntax of the abstract data structures representing managed object values.

6.2.2 Textual conventions

```
NetworkType ::= INTEGER {
    ipv4 (1),
    ipv6 (2),
    asi (3)
} (ipv4..asi)
-- An enumeration identifying a network type of over which the
-- media is flowing.

TransportType ::= INTEGER {
    notApplicable (0),
    rtp (1)
} (notApplicable.. rtp)
-- 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

AudioFECType ::= INTEGER {
    none(0),
    rfc2733(1),
    rfc5109(2),
    smpte2022Dash1(3),
    smpte2022Dash5(4),
    proprietary(5)
} (none..proprietary)
-- An enumeration identifying the FEC type applied, if present.

VideoFECType ::= INTEGER {
    none (0),
    rfc2733(1),
    rfc5109(2),
    smpte2022Dash1(3),
    smpte2022Dash5(4),
    proprietary(5)
} (none..proprietary)
-- An enumeration identifying the FEC type applied, if present.

BufferSize ::= Unsigned32
-- A type to indicate the current total size of the receive buffer
-- in ms.

BufferOcpncyTime ::= Gauge32
-- A type to report the amount of data, expressed in ms,
-- occupying the receive buffer.

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

TemperatureLocn ::= OCTET STRING (0..80)
```

-- A type to indicate the location where the temperature is measured.

TemperatureTrend ::= Gauge32

-- 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.

TemperatureStatus ::= INTEGER {

undetermined (0),

other (1),

unknown (2),

ok (3),

warning (4),

critical (5),

nonRecoverable (6)

} (undetermined.. nonRecoverable)

-- An enumeration identifying the temperature status levels.

-- Semantics are equipment specific.

BitRateType ::= INTEGER {

unspecified (0),

vbr (1),

cbr (2)

} (unspecified..cbr)

-- An enumeration identifying the video bit rate type applied

--

-- vbr = variable bit rate

-- cbr = constant bit rate

6.2.3 Sequences

nMtEntry ::= SEQUENCE {

nMtBlockId BlockId,

nMtIfIndex InterfaceIndex,

nMtTxRxPoint TruthValue,

nMtNetworkType NetworkType,

nMtTransportType TransportType,

nMtTxRxAddr TAddress,

nMtPortNumber CardinalNumber,

nMtIGMPVersion CardinalNumber,

nMtSIPServerAddr TAddress

}

aMtBlockEntry ::= SEQUENCE {

aMtBlockId BlockId,

aMtAudioComponentNumber IndexNumber,

aMtNetworkBlockId BlockId,

aMtAudioStatus TruthValue,

aMtAudioSignalFormat MediaFormat,

aMtAudioPID CardinalNumber,

aMtIfIndex InterfaceIndex,

aMtFECType AudioFECType,

aMtFECLengthDimension IntegerNumber

}

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
}

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

temperatureEntry ::= SEQUENCE {
    temperatureBlockId      BlockId,
    temperatureLocnNumber   IndexNumber,
    temperatureLocation     TemperatureLocn,
    temperatureTrend        TemperatureTrend,
    temperatureStatus       TemperatureStatus,
    temperatureLowWarning   IntegerNumber,
    temperatureHighWarning  IntegerNumber,
    temperatureLowCritical  IntegerNumber,
    temperatureHighCritical IntegerNumber
}

```

6.3 Network measurement information blocks

6.3.1 Network measurement information block structure

A network measurement information block, shown in Figure 2, shall have the following structure, where *c* is the number of channels on a connection.

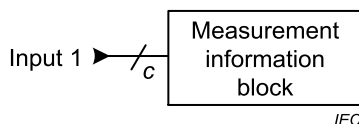


Figure 2 – Network measurement information block

A network measurement information block shall consist of a generic measurement information block containing information specific to network parameters that are to be available for use by a management agent.

NOTE This is a special type of block that has no 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 standard does not define where the information for the content of these blocks is sourced from.

The group of objects in Table 1 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 1 – 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

6.3.2 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.

6.3.3 nMtBlockEntry

An entry in the network measurement block table.

6.3.4 nMtBlockId

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

6.3.5 nMtIfIndex

This identifier provides a link between this network measurement block and the network interface it is associated with. The values of this object shall be a copy of the values of ifIndex from RFC1213-MIB (MIB-II) (1.3.6.1.2.1.2.2.1.1).

NOTE The nMtIfIndex object identifier applies the semantics of ifIndex through the use of the InterfaceIndex textual convention defined in RFC2863, The Interfaces Group MIB, instead of using the original syntax of ifIndex in RFC1213-MIB (MIB-II), which is the identifier for the associated network interfaces object within MIB-II.

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 and shall remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

6.3.6 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, this value should be set to indicate the position from where the measurement is required to be made, that is 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, 6.6.2) shall exist, otherwise its existence is optional.

6.3.7 nMtNetworkType

Describes the current network type over which the media 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.

6.3.8 nMtTransportType

Describes the transport type being used to convey the media.

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

6.3.9 nMtTxRxAddr

The network address of either the send or receive point network interface over which the media is flowing. Which it is, depends on the measurement position in the media chain, that is 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.

6.3.10 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.

6.3.11 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.

6.3.12 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.

6.4 Audio measurement information blocks

6.4.1 Audio measurement information block structure

An audio measurement information block, as shown in Figure 3, shall have the following structure, where c is the number of channels on a connection.

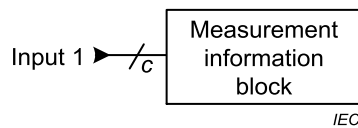


Figure 3 – Audio measurement information block

An audio measurement information block shall consist of a generic measurement information block containing information specific to audio parameters that are to be available for use by a management agent.

NOTE This is a special type of block that has no 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, through the input and for this information to be used and presented 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.

The group of objects in Table 2 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 2 – 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

6.4.2 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.

6.4.3 aMtBlockEntry

An entry in the audio measurement block table.

6.4.4 aMtBlockId

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

6.4.5 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 6.4.8 object.

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

6.4.6 aMtNetworkBlockId

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

6.4.7 aMtAudioStatus

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

6.4.8 aMtAudioSignalFormat

Describes 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 `aMtAudioStatus` is `false`, the value `noAudio` shall be returned.

6.4.9 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.

6.4.10 aMtIfIndex

This identifier provides a link between this audio component and the network interface on which it is present. The values of this object shall be a copy of the values of `ifIndex` from RFC1213-MIB (MIB-II) (1.3.6.1.2.1.2.2.1.1).

NOTE The `aMtIfIndex` object identifier applies the semantics of `ifIndex` through the use of the `InterfaceIndex` textual convention defined in RFC2863, The Interfaces Group MIB, instead of using the original syntax of `ifIndex` in RFC1213-MIB (MIB-II), which is the identifier for the associated network interfaces object within MIB-II.

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` and shall remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

6.4.11 `aMtFECType`

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

6.4.12 `aMtFECLengthDimension`

Describes 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 `xyyy`, where `xx` represents the number of bytes over which FEC is applied and `yy` represents the matrix size, for example, 2405 for 24 B and a 5×5 matrix.

6.5 Video measurement information blocks

6.5.1 Video measurement information block structure

A video measurement information block, as shown in Figure 4, shall have the following structure, where `c` is the number of channels on a connection.

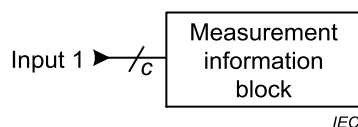


Figure 4 – Video measurement information block

A video measurement information block shall consist of a generic measurement information block containing information specific to video parameters that are to be available for use by a management agent.

NOTE This is a special type of block that has no 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 standard does not define where the information for the content of these blocks is sourced from.

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) videoMeasurement(3) }
```

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

Table 3 – Managed objects for video measurement information blocks

Identifier	Syntax	Index	Readable	Writable	Volatile	Status
vMtBlockTable(1)	SEQUENCE OF VMtBlockEntry		none	none	no	m
LvMtBlockEntry(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
LvMtTrickModeSupport(12)	TruthValue		listener	none	yes	o

6.5.2 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.

6.5.3 vMtBlockEntry

An entry in the video measurement block table.

6.5.4 vMtBlockId

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

6.5.5 vMtAudioBlockId

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

6.5.6 vMtNetworkBlockId

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

6.5.7 vMtVideoStatus

The status of the video signal.

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

6.5.8 vMtVideoSourceFormat

Describes the structure of the base video source format.

The format is defined in IEC 62379-3.

6.5.9 vMtVideoCodingType

Describes the current video signal coding type.

The format is defined in IEC 62379-3.

6.5.10 vMtVideoBitRateType

Describes the current video bit rate type that is applied.

6.5.11 vMtVideoBitRate

Describes 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.

6.5.12 vMtVideoAspectRatio

Describes the current video aspect ratio.

The format is defined in IEC 62379-3.

6.5.13 vMtFECType

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

6.5.14 vMtFECLengthDimension

Describes 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 B and a 5×5 matrix.

6.5.15 vMtTrickModeSupport

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

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

6.6 Receiver point measurement information block

6.6.1 Receiver measurement information block structure

A receiver point measurement information block, as shown in Figure 5, shall have the following structure, where `c` is the number of channels on a connection.

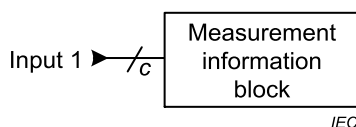


Figure 5 – Receiver measurement information block

A receiver measurement information block shall consist of a generic measurement information block containing information specific to receiver parameters that are to be available for use by a management agent.

NOTE This is a special type of block that has no 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 standard does not define where the information for the content of these blocks is sourced from.

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) receiverMeasurement(4) }

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 4 – Managed objects for receiver measurement information blocks

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

6.6.2 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`.

6.6.3 `rxPointEntry`

An entry in the receiver point measurement block table.

6.6.4 `rxPointBlockId`

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

6.6.5 `rxPointNetworkBlockId`

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

6.6.6 `rxPointBufferSize`

Describes the current total size of the receive buffer in ms.

6.6.7 `rxPointBufferOcpancyTime`

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

6.6.8 `rxPointBufferOcpncyPcnt`

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

6.6.9 `rxPointMDI`

Describes 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 values in the form of DF:MLR, e.g. 06:10.

The measurement of DF and MLR is in ms.

6.6.10 `rxPointTSDF`

Describes the time-stamped delay factor (TS-DF).

Time-stamped delay factor is calculated as:

$$\text{TS-DF} = D_{\max} - D_{\min}$$

The measurement of TS and DF is in ms.

For further information, see EBU-Tech 3337.

6.7 Temperature measurement information block

6.7.1 Temperature measurement information block structure

A temperature measurement information block, as shown in Figure 6, shall have the following structure, where *c* is the number of channels on a connection.

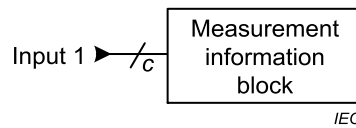


Figure 6 – Temperature measurement information block

A temperature measurement information block shall consist of a generic measurement information block containing information specific to temperature parameters that are to be available for use by a management agent.

NOTE This is a special type of block that has no 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 standard does not define where the information for the content of these blocks is sourced from.

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)
  temperatureMeasurement(5) }
```

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

Table 5 – 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

6.7.2 temperatureTable

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

6.7.3 temperatureEntry

An entry in the temperature measurement table.

6.7.4 temperatureBlockId

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

6.7.5 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.

The temperature measurement location numbers should be allocated sequentially, starting from 1.

6.7.6 temperatureLocation

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

6.7.7 temperatureTrend

The current temperature measured at this location, in degrees Celsius (°C).

6.7.8 temperatureStatus

Describes the current status of this particular temperature measurement, as follows:

- `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.

6.7.9 temperatureLowWarning

Describes the low warning limit for the measured temperature, in degrees Celsius (°C).

6.7.10 temperatureHighWarning

Describes the high warning limit for the measured temperature, in degrees Celsius (°C).

6.7.11 temperatureLowCritical

Describes the low critical limit for the measured temperature, in degrees Celsius (°C).

6.7.12 temperatureHighCritical

Describes the high critical limit for the measured temperature, in degrees Celsius (°C).

Annex A (informative)

Machine-readable measurement block definitions

This annex provides the measurement block definitions in a machine-readable format interpretable by standard MIB browsing software tools. It does not express all the requirements of this standard, for instance where access to an object is restricted at certain privilege levels. If there is any inconsistency between this annex and Clause 6, the latter takes 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 "201506041200Z"      -- June 04, 2015 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 "201506041200Z"      -- June 04, 2015 at 12:00 GMT
```

```
    DESCRIPTION
```

```
        "Expanded and re-ordered
        AudioFECType and VideoFECType
        entries due to previous
        errors and missing types.
        Clarified description of
        nMtIfIndex and aMtIfIndex."
```

```
    REVISION "201312031200Z"      -- December 03, 2013 at 12:00 GMT
```

```
    DESCRIPTION
```

```
        "Corrected various errors.
        Limited width of descriptions
        in various objects to ensure
        that copies of the .my files
        formatted correctly in the
```

```

standards documents."
REVISION "201106211200Z"    -- June 21, 2011 at 12:00 GMT
DESCRIPTION
  "Added additional coding types VP8 and H.264 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 62379-7:2013."
  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
    "This identifier provides a link between this network
    measurement block and the network interface it is
    associated with. The values of this object shall be a copy
    of the values of ifIndex from RFC1213-MIB (MIB-II)
    (1.3.6.1.2.1.2.2.1.1).

```

Note The nMtIfIndex object identifier applies the semantics of ifIndex through the use of the InterfaceIndex textual convention defined in RFC2863, The Interfaces Group MIB, instead of using the original syntax of ifIndex in RFC1213-MIB (MIB-II), which is the identifier for the associated network interfaces object within MIB-II.

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 and shall remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

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 ifIndex from RFC2863, The Interfaces Group MIB:

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

Description of Textual Convention InterfaceIndex from RFC2863, The Interfaces Group 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."

```
::= { 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.
        Whichever it is, it 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, this indicates that the audio signal is present.
        If false, this indicates that 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 aMtAudioStatus 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
        "This identifier provides a link between this audio
        component and the network interface it is
        associated with. The values of this object shall be a copy
        of the values of ifIndex from RFC1213-MIB (MIB-II)
        (1.3.6.1.2.1.2.2.1.1)."
```

Note The aMtIfIndex object identifier applies the semantics of ifIndex through the use of the InterfaceIndex textual convention defined in RFC2863, The Interfaces Group MIB, instead of using the original syntax of ifIndex in RFC1213-MIB (MIB-II), which is the identifier for the associated network interfaces object within MIB-II.

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 and shall remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

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 ifIndex from RFC2863, The Interfaces Group MIB:

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

Description of Textual Convention InterfaceIndex from RFC2863, The Interfaces Group 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 Forward Error Correction (FEC) type
        applied, if present.

        Current Values are:

        none(0),
        rfc2733(1),
        rfc5109(2),
        smpte2022Dash1(3),
        smpte2022Dash5(4),
        proprietary(5)"
    ::= { 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 IEC62379-3
        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 IEC62379-3
        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 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"
    ::= { 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 IEC62379-3
        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
vMtFEType OBJECT-TYPE
    SYNTAX VideoFEType
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "An indication as to the Forward Error Correction (FEC) type
        applied, if present.

        Current Values are:

        none(0),

```

```

    rfc2733(1),
    rfc5109(2),
    smpte2022Dash1(3),
    smpte2022Dash5(4),
    proprietary(5) "
::= { 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.

        If true, this indicates trick mode is supported.
        If false, this indicates trick mode is not supported."
::= { 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,
        rxPointBufferOcpncyTime
            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
rxPointBufferOcpncyTime 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 about this, see EBU TECH3337"
    ::= { rxPointEntry 7 }

-- 1.0.62379.7.1.4.3
receiverMeasurementGroup OBJECT-GROUP
    OBJECTS { rxPointBufferSize, rxPointBufferOcpncyTime,
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
        "A row in the temperature table. Entries cannot
        be created or deleted with SNMP operations."
```

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 "Degree Celsius"
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The current temperature measured at this location.
        The unit is the degree 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 "Degree Celsius"
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The low warning limit for the measured temperature.
        The unit is the degree Celsius."
    ::= { temperatureEntry 6 }

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

```

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

-- 1.0.62379.7.1.5.1.1.9
temperatureHighCritical OBJECT-TYPE
    SYNTAX IntegerNumber
    UNITS "Degree Celsius"
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The high critical limit for the measured temperature.
        The unit is the degree 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 B (informative)

Machine-readable textual conventions definitions

This annex provides the textual conventions definitions in a machine-readable format interpretable by standard MIB browsing software tools. If there is any inconsistency between this annex and 6.2.2, the latter takes 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 "201506041200Z"      -- June 04, 2015 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 "201506041200Z"      -- June 04, 2015 at 12:00 GMT
    DESCRIPTION
        "Expanded and re-ordered
        AudioFECType and VideoFECType
        entries due to previous
        errors and missing types."
    REVISION "201312031200Z"      -- December 03, 2013 at 12:00 GMT
    DESCRIPTION
        "Corrected various errors.
        Limited width of descriptions
        in various objects to ensure
        that copies of the .my files
        formatted correctly in the
        standards documents."
    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),
        rfc2733(1),
        rfc5109(2),
        smpte2022Dash1(3),
        smpte2022Dash5(4),
        proprietary(5)
    }
-- {
-- none(0),
-- rfc2733(1),
-- rfc5109(2),
-- smpte2022Dash1(3),
-- smpte2022Dash5(4),
-- proprietary(5)
-- } (none..proprietary)
--
VideoFECType ::= TEXTUAL-CONVENTION
    STATUS current
    DESCRIPTION
        "An enumeration identifying the FEC type applied, if present."
    SYNTAX INTEGER
    {
        none(0),
        rfc2733(1),
        rfc5109(2),
        smpte2022Dash1(3),
        smpte2022Dash5(4),
        proprietary(5)
    }
-- {
-- none(0),

```

```
-- rfc2733(1),
-- rfc5109(2),
-- smpte2022Dash1(3),
-- smpte2022Dash5(4),
-- proprietary(5)
-- } (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 C (informative)

Worked example

C.1 Overview

This annex provides a worked example to show how Parts 1, 2, 3 and 7 of IEC 62379 series may be used in a vendor's MIB to standardise the measurement methods required by the EBU ECN-IPM Group. If there is any inconsistency between this annex and the main clauses of Parts 1, 2, 3 and 7, those clauses take precedence.

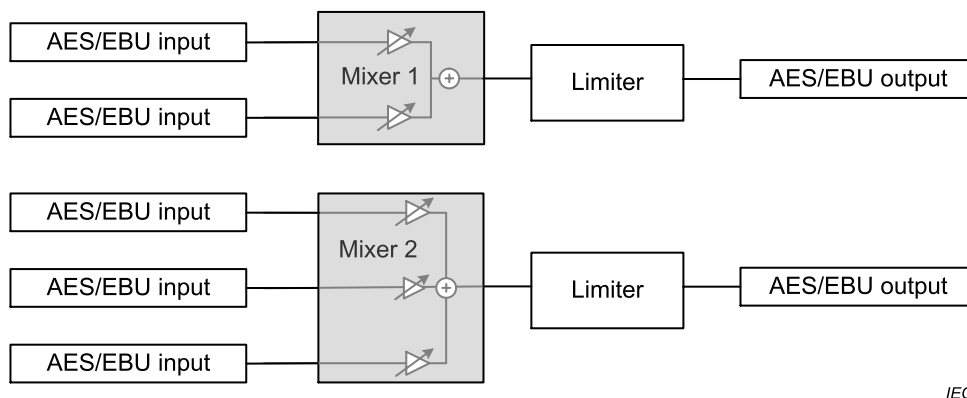
The examples shown here are intended to provide only a high level view of the process involved. Annex E and F of IEC 62379-2:2008 provide detailed examples of how the block structure may be used to model audio devices. Similar processes may be applied to model video devices, see IEC 62379-3.

C.2 Example 1

C.2.1 General

To show the relationships between the various tables defined in Parts 1, 2, 3 and 7 of IEC 62379 the example shown here is a modified version of example 1 in Annex E of IEC 62379-2:2008, to model an audio device which is then extended to describe a device that has multiple encoders/decoders of different types present in it and where there are multiple measurement blocks present.

The modified audio example device has two mixers, one with two inputs, the other with three, providing in addition, a limiter for each, as represented by Figure C.1.



IEC

Figure C.1 – Example of a modified audio device

NOTE This unit example could be built using a single mixer block with five inputs and two outputs, but this arrangement is more appropriate to illustrate the use of measurement blocks.

Details of the audio that might be carried are not relevant for the description of this example, so are not included.

C.2.2 Block table

First, a block table is required showing all the functional blocks that make up the unit. The mapping of a block to a blockId is arbitrary. It is essential that a particular block is always associated with the same blockId.

The table is located at `iso.standard.iec67329.general.generalMIB.block.blockTable (1.0.62379.1.1.2.1)`.

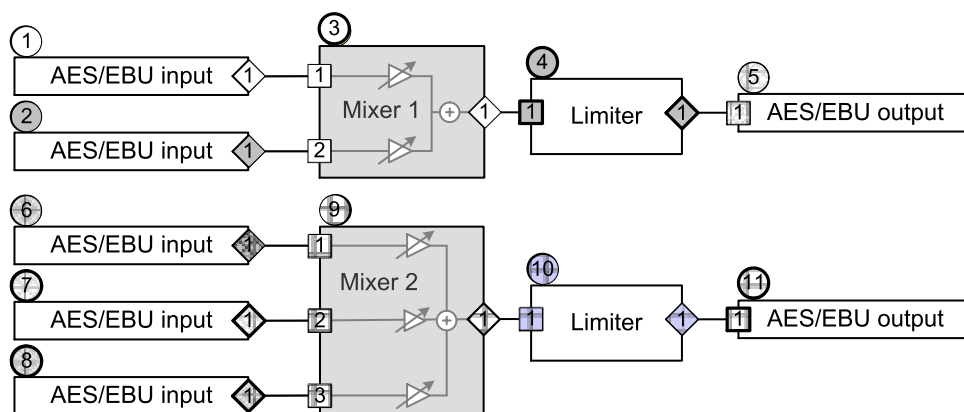
As there are eleven blocks in the example, this table will have eleven rows.

Table C.1 – Main block Id table

.blockTable.blockEntry (.1.1)	
<i>blockId (1)</i>	<i>blockType (2)</i>
①	1.0.62379.2.1.1 (Audio port)
②	1.0.62379.2.1.1 (Audio port)
③	1.0.62379.2.1.2 (Mixer block)
④	1.0.62379.2.1.5 (Limiter block)
⑤	1.0.62379.2.1.1 (Audio port)
⑥	1.0.62379.2.1.1 (Audio port)
⑦	1.0.62379.2.1.1 (Audio port)
⑧	1.0.62379.2.1.1 (Audio port)
⑨	1.0.62379.2.1.2 (Mixer block)
⑩	1.0.62379.2.1.5 (Limiter block)
⑪	1.0.62379.2.1.1 (Audio port)

The above table, Table C.1, identifies each of the block types that exist within a unit, along with the unique identifier [*blockId (1)*]. Although the blockIds are shown as incrementing numbers, any positive integer number can be used and in any order, as long as each is unique.

In example 1 of Annex E of IEC 62379-2:2008, the structure of the unit is shown with a connector table to express all the links that exist between the blocks defined in the block table. The connector table contains each block's blockId (in circles), input numbers (squares) and output numbers (diamonds) and the diagram of the unit now looks like the one shown in Figure C.2, annotated with these shaped numbers.



IEC

Figure C.2 – Annotated connector diagram

NOTE The connector table contents are not necessary for describing the use of the measurement blocks, nor are the contents or description of either the mode table or the audio port tables, as shown in example 1 of Annex E of IEC 62379-2:2008, although all would be required for a real implementation of the extended example shown here.

For clarity, only the mixer section of the above diagram will now be used, so that the structure of an audio device having these two mixer blocks will now look like the diagram below, Figure C.3, with the related block table, Table C.2.

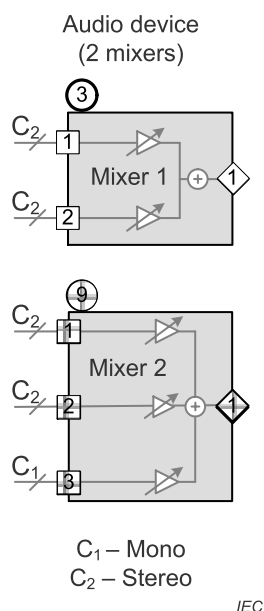


Figure C.3 – Mixer section

Table C.2 – Mixer related block Id table

.blockTable.blockEntry (.1.1)	
blockId (1)	blockType (2)
①	1.0.62379.2.1.1 (Audio port)
②	1.0.62379.2.1.1 (Audio port)
③	1.0.62379.2.1.2 (Mixer block)
④	1.0.62379.2.1.5 (Limiter block)
⑤	1.0.62379.2.1.1 (Audio port)
⑥	1.0.62379.2.1.1 (Audio port)
⑦	1.0.62379.2.1.1 (Audio port)
⑧	1.0.62379.2.1.1 (Audio port)
⑨	1.0.62379.2.1.2 (Mixer block)
⑩	1.0.62379.2.1.5 (Limiter block)
⑪	1.0.62379.2.1.1 (Audio port)

Not shown
for clarity

Not shown
for clarity

Not shown
for clarity

The two mixer blocks, limiter blocks and audio port blocks are standard blocks defined in IEC 62379-2. The mixer inputs (and outputs) can be stereo, mono or other channel arrangements as defined in the audio signal formats.

C.2.3 Mixer block

The mixer block is rooted at iso.standard.iec62379.audio.audioMIB.audioMixer (1.0.62379.2.1.2).

This block consists of two tables, Table C.3, aMixerBlockTable, located at 1.0.62379.2.1.2.1 and aMixerInputTable, located at 1.0.62379.2.1.2.2. There is one entry in aMixerBlockTable for each mixer block in a unit and an entry in aMixerInputTable for each input of each mixer.

Table C.3 – Mixer block tables

.aMixerBlockTable.aMixerBlockEntry (.1.1)				
<i>aMixerBlockId</i> (1)	<i>aMixerFadeDuration</i> (2)	<i>aMixerFadeNow</i> (3)		
③				
⑨				

.aMixerInputTable.aMixerInputEntry (.2.1)				
<i>aMixerInputBlockId</i> (1)	<i>aMixerInputNumber</i> (2)	<i>aMixerInputLevel</i> (3)	<i>aMixerInputFadeToLevel</i> (4)	<i>aMixerInputDelay</i> (5)
③	1			
③	2			
⑨	1			
⑨	2			
⑨	3			

As the example has two mixer blocks, one with two inputs, the other with three, `aMixerBlockTable` has two entries (of type `AMixerBlockEntry`). This defines the overall fader information for each mixer block.

The larger of the two tables, `aMixerInputTable` has five entries (of type `AMixerInputEntry`), two for mixer block ID 3 and three for mixer block ID 9. The number of times these exist being dependent upon the number of inputs in each mixer block. This allows each input of each mixer block to be uniquely identified.

The remaining entries in this table would contain information relevant to each input.

NOTE Other information pertaining to these mixer blocks is not included, such as the connections to other parts of the initial example. This information would be held in other tables, such as connection and/or mode tables, similar to the example shown in IEC 62379-2.

C.2.4 Multiple functionality device

C.2.4.1 General

Having considered how the process of assigning blockIds and input identifiers describes a unit, a similar principle can be applied in this example, where we consider a single device with multiple functionality within it, consisting of what we have already used (2 mixer blocks) and now adding 2 network interfaces, 1 audio codec and 3 video codecs, each with a differing number of associated audio components, Figure C.4 and Figure C.5.

Each type of standard measurement block has an entry in the main block table, appearing as many times as there are codecs or network interfaces, each with a unique identifier (blockId). Measurement blocks of the same type all have the same OID, i.e. all audio measurement blocks have the same OID, but different blockIds, as shown in Table C.4.

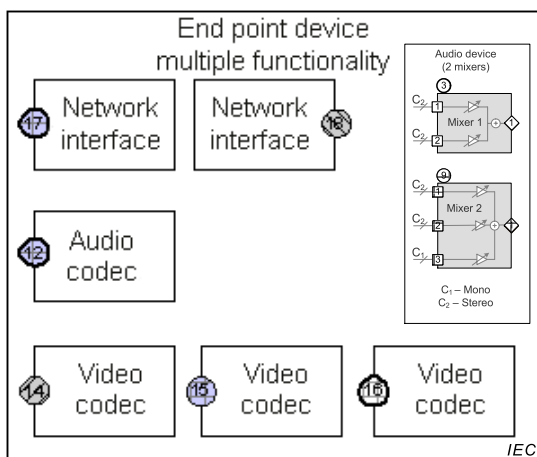


Figure C.4 – Single device with multiple functionality

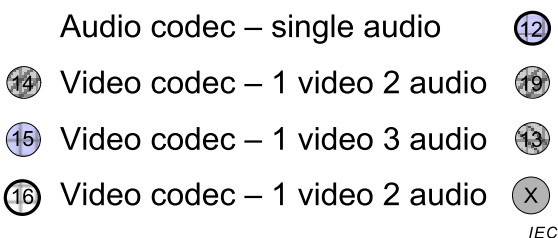


Figure C.5 – Measurement blockIds and their associated media components

Table C.4 – Addition of measurement block Ids

.blockTable.blockEntry (.1.1)	
blockId (1)	blockType (2)
①	1.0.62379.2.1.1 (Audio port)
②	1.0.62379.2.1.1 (Audio port)
③	1.0.62379.2.1.2 (Mixer block)
④	1.0.62379.2.1.5 (Limiter block)
⑤	1.0.62379.2.1.1 (Audio port)
⑥	1.0.62379.2.1.1 (Audio port)
⑦	1.0.62379.2.1.1 (Audio port)
⑧	1.0.62379.2.1.1 (Audio port)
⑨	1.0.62379.2.1.2 (Mixer block)
⑩	1.0.62379.2.1.5 (Limiter block)
⑪	1.0.62379.2.1.1 (Audio port)
⑫	1.0.62379.7.1.1 (Audio measurement)
⑬	1.0.62379.7.1.1 (Audio measurement)
⑭	1.0.62379.7.1.2 (Video measurement)
⑮	1.0.62379.7.1.2 (Video measurement)
⑯	1.0.62379.7.1.2 (Video measurement)
⑰	1.0.62379.7.1.3 (Network measurement)
⑱	1.0.62379.7.1.3 (Network measurement)
⑲	1.0.62379.7.1.1 (Audio measurement)
X	1.0.62379.7.1.1 (Audio measurement)

The blockIds in Figure C.5 are for the different types of measurement blocks only. In practice there would be additional blockId entries for the codecs themselves and whatever facilities that these provided within the equipment, similar to the audio mixer blocks in Figure C.3.

For example, an audio codec might be constructed from one or more standard audio blocks specified in IEC 62379-2 (or elsewhere), such as a converter block, so this would also have an entry in the main block table.

C.2.4.2 Audio components

Having now defined each measurement block ID, it is necessary to deal with the various audio components associated with each measurement block. The audio measurement table is used commonly for both audio only codecs and all the audio components associated with the video codecs. The audio measurement blockIds associated with each video codec, shown in Figure C.5, can be added to the overall multifunctional device, see Figure C.6.

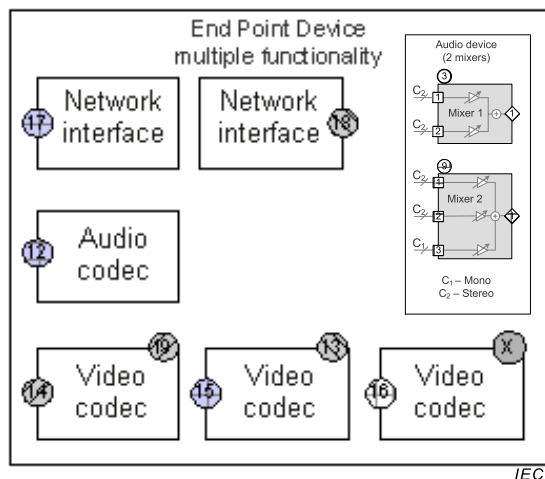


Figure C.6 – Single device with multiple functionality

Table C.5 – Video measurement table

Video measurement table			
	<i>vMtAudio BlockId (2)</i>	<i>vMtNetwork BlockId (3)</i>	<i>Other video parameters</i>
14	19		
15	13		
16	X		

Table C.6 – Network measurement table

Network measurement table	
<i>nMtBlockId (1)</i>	<i>Other network parameters</i>
17	
18	

Table C.7 – Audio measurement table

Audio measurement table			
<i>aMtBlockId (1)</i>	<i>aMtAudioComponent Number (2)</i>	<i>aMtNetwork BlockId (3)</i>	<i>Other audio parameters</i>
12	1		
13	1		
13	2		
13	3		
19	1		
19	2		
X	1		
X	2		

Audio codec (ID 12) has only one audio associated with it, so there is only one entry line in the audio measurement table, see Table C.7. Video codec (ID 14) has 2 audio components. It has one entry in the video measurement table, see Table C.5 and an associated audio measurement block ID of 19, see Figure C.6. This is used to identify the row(s) in the audio measurement table that are the associated audio components with this video. The audio measurement table, Table C.7, has therefore 2 entries for the audio components with ID 19 and the audio component number is used to uniquely identify each audio component.

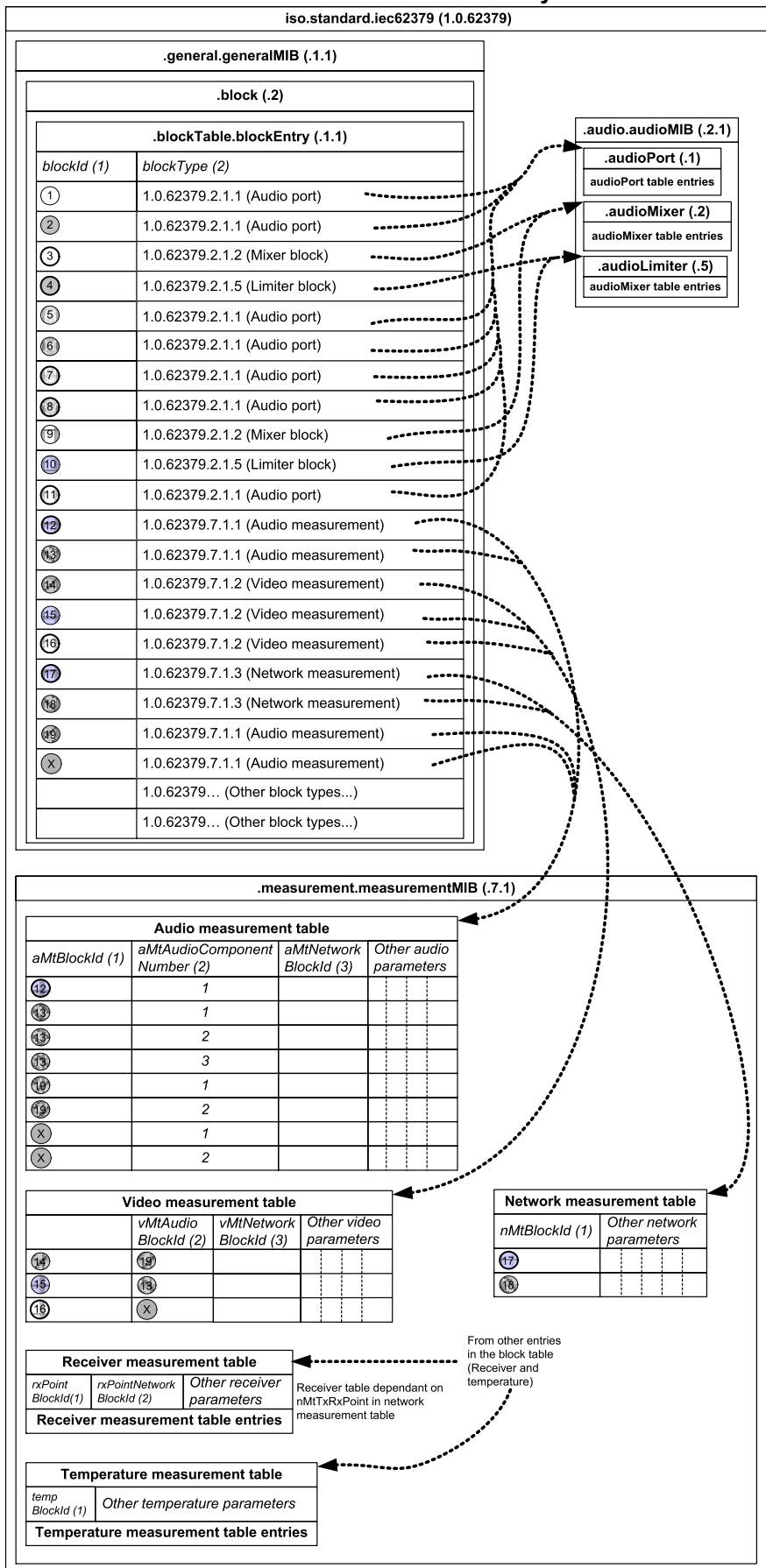
Although not completed in this example, whichever network the codecs might be connected to internally, the appropriate network ID would need to be entered into the audio and video measurement tables matching up with the correct audio or video codec block ID.

A similar process is used for all the other video components and their associated audio components. Each network interface has IDs of their own in the network measurement table, see Table C.6.

This whole process allows for devices with multiple functionalities, where the functionality may be different or repeated many times.

C.2.5 Summary of tables

Table C.8 – Table summary



Bibliography

IEC 62379 (all parts) *Common control interface for networked audio and video product*

EBU-TECH 3337, *A Proposed Time-Stamped Delay Factor (TS-DF) algorithm for measuring Network Jitter on RTP Streams*, Geneva, January 2010. Available from <<http://tech.ebu.ch/webdav/site/tech/shared/tech/tech3337.pdf>>

EBU-TECH 3345, *End-to-End IP Network Measurement for Broadcast Applications – Parameters & Management Information Base (MIB)*, Geneva, July 2011. Available from <<http://tech.ebu.ch/webdav/site/tech/shared/tech/tech3346.pdf>>

EBU-Tech 3346, *End-to-End IP Network Measurement for Broadcast Applications – EisStream Software package description*, Geneva, July 2011. Available from <<http://tech.ebu.ch/webdav/site/tech/shared/tech/tech3346.pdf>>

IETF STD 58 (SMIPv2). RFC 2578 McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIPv2)", STD 58, RFC 2578, April 1999. Available from <<https://ietf.org/rfc/rfc2578.txt>>

RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II*
available from <https://www.ietf.org/rfc/rfc1213.txt>

RFC 2733, *An RTP Payload Format for Generic Forward Error Correction*
available from <https://www.ietf.org/rfc/rfc2733.txt>

SMPTE ST 2022-1:2007, *Forward Error Correction for Real-Time Video/Audio Transport Over IP Networks*
available from <http://standards.smpte.org>

SMPTE ST 2022-5:2013, *Forward Error Correction for Transport of High Bit Rate Media Signals over IP Networks (HBRMT)*
available from <http://standards.smpte.org>

SMPTE ST 2016-1:2009, *Format for Active Format Description and Bar Data*
available from <http://standards.smpte.org>

SMPTE ST 2042-1:2012, VC-2, *VC-2 Video Compression*
available from <http://standards.smpte.org>

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

3, rue de Varembé
PO Box 131
CH-1211 Geneva 20
Switzerland

Tel: + 41 22 919 02 11
Fax: + 41 22 919 03 00
info@iec.ch
www.iec.ch