



ISO/IEC 29341-17-11

Edition 1.0 2011-08

INTERNATIONAL STANDARD



**Information technology – UPnP device architecture –
Part 17-11: Quality of Service Device Control Protocol – Level 3 – Quality of
Service Manager Service**



THIS PUBLICATION IS COPYRIGHT PROTECTED
Copyright © 2011 ISO/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 ISO/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
Email: inmail@iec.ch
Web: 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.

- Catalogue of IEC publications: www.iec.ch/searchpub

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 It also gives information on projects, withdrawn and replaced publications.

- IEC Just Published: www.iec.ch/online_news/justpub

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

- Electropedia: www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 20 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary online.

- Customer Service Centre: www.iec.ch/webstore/custserv

If you wish to give us your feedback on this publication or need further assistance, please visit the Customer Service Centre FAQ or contact us:

Email: csc@iec.ch
Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00



ISO/IEC 29341-17-11

Edition 1.0 2011-08

INTERNATIONAL STANDARD



**Information technology – UPnP device architecture –
Part 17-11: Quality of Service Device Control Protocol – Level 3 – Quality of
Service Manager Service**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

PRICE CODE



ICS 35.200

ISBN 978-2-88912-654-5

CONTENTS

1	Overview and Scope.....	3
1.1	References & Terms.....	3
1.1.1	References.....	3
1.1.2	Terms and Abbreviations.....	5
2	Service Modeling Definitions.....	12
2.1	ServiceType.....	12
2.2	State Variables.....	12
2.2.1	XML Fragments as UPnP Arguments.....	12
2.2.2	A_ARG_TYPE_TrafficDescriptor.....	13
2.2.3	A_ARG_TYPE_TrafficHandle.....	33
2.2.4	A_ARG_TYPE_NumTrafficDescriptors.....	33
2.2.5	A_ARG_TYPE_NumPolicyHolders.....	33
2.2.6	A_ARG_TYPE_ListOfTrafficDescriptors.....	33
2.2.7	A_ARG_TYPE_QmCapabilities.....	35
2.2.8	A_ARG_TYPE_ExtendedTrafficQosInfo.....	36
2.2.9	A_ARG_TYPE_ResultedQosType.....	38
2.2.10	Relationships Between State Variables.....	38
2.3	Eventing and Moderation.....	39
2.3.1	Event Model.....	39
2.4	Actions.....	39
2.4.1	RequestTrafficQos().....	39
2.4.2	UpdateTrafficQos().....	51
2.4.3	ReleaseTrafficQos().....	59
2.4.4	BrowseAllTrafficDescriptors().....	59
2.4.5	GetQmCapabilities().....	60
2.4.6	RequestExtendedTrafficQos().....	61
2.4.7	UpdateExtendedTrafficQos().....	66
2.4.8	Common Error Codes.....	70
2.5	Theory of Operation (Informative).....	72
2.5.1	Overview.....	72
2.5.2	Recovery from Failures.....	73
3	XML Service Description.....	75
4	Test.....	79
	Annex A (normative) Traffic Descriptor Matrix.....	80
	Annex B (informative) Methods for determining candidate streams for preemption based on UINS.....	83
	Annex C (normative) The <i>res@tspec</i> property in the <i>ContentDirectory</i> Service.....	84
	C.1 Definition of the XML Fragment.....	84
	Figure 1 — An example traffic stream.....	22
	Figure 2 — A traffic stream with a constant bit rate.....	24
	Figure 3 — The non-constant bit-rate traffic (left) can be viewed as a constant bit-rate traffic of a higher constant bit-rate <i>D</i> (right).....	24

Table 1-1 — Abbreviations.....	5
Table 2-1 — State Variables	13
Table 2-2 — Meaning of <u>PolicyHolderConsultedType</u> parameter.....	18
Table 2-3 — Traffic Specification Parameters.....	21
Table 2-4 — Meaning of <u>RequestedQosType</u> parameter.....	22
Table 2-5 — Meaning of <u>ResultedQosType</u> parameter.....	38
Table 2-6 — Event Moderation.....	39
Table 2-7 — Actions	39
Table 2-8 — Arguments for <u>RequestTrafficQos()</u>	40
Table 2-9 — Elements that MUST equal those of the original TrafficDescriptor	42
Table 2-10 — Default Priorities used by the <u>QoS Manager</u> based on 802.1D (Annex G).....	44
Table 2-11 — Error Codes for <u>RequestTrafficQos()</u>	50
Table 2-12 — Elements that are allowed to be updated when calling <u>UpdateTrafficQos()</u>	52
Table 2-13 — Arguments for UpdateTrafficQos()	53
Table 2-14 — Error Codes for UpdateTrafficQos	58
Table 2-15 — Arguments for ReleaseTrafficQos.....	59
Table 2-16 — Error Codes for ReleaseTrafficQos.....	59
Table 2-17 — Arguments for BrowseAllTrafficDescriptors.....	60
Table 2-18 — Error Codes for BrowseAllTrafficDescriptors.....	60
Table 2-19 — Arguments for GetQmCapabilities	60
Table 2-20 — Error Codes for GetQmCapabilities	61
Table 2-21 — Arguments for RequestExtendedTrafficQos	61
Table 2-22 — Error Codes for RequestExtendedTrafficQos	65
Table 2-23 — Arguments for UpdateExtendedTrafficQos.....	66
Table 2-24 — Error Codes for UpdateExtendedTrafficQos.....	69
Table 2-25 — Common Error Codes	70

INFORMATION TECHNOLOGY – UPNP DEVICE ARCHITECTURE –

Part 17-11: Quality of Service Device Control Protocol – Level 3 – Quality of Service Manager Service

FOREWORD

- 1) ISO (International Organization for Standardization) and IEC (International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards. Their preparation is entrusted to technical committees; any ISO and IEC member body interested in the subject dealt with may participate in this preparatory work. International governmental and non-governmental organizations liaising with ISO and IEC also participate in this preparation.
- 2) In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.
- 3) The formal decisions or agreements of IEC and ISO 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 and ISO member bodies.
- 4) IEC, ISO and ISO/IEC publications have the form of recommendations for international use and are accepted by IEC and ISO member bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC, ISO and ISO/IEC publications is accurate, IEC or ISO cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 5) In order to promote international uniformity, IEC and ISO member bodies undertake to apply IEC, ISO and ISO/IEC publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any ISO/IEC publication and the corresponding national or regional publication should be clearly indicated in the latter.
- 6) ISO and IEC provide no marking procedure to indicate their approval and cannot be rendered responsible for any equipment declared to be in conformity with an ISO/IEC publication.
- 7) All users should ensure that they have the latest edition of this publication.
- 8) No liability shall attach to IEC or ISO or its directors, employees, servants or agents including individual experts and members of their technical committees and IEC or ISO member bodies 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 of, use of, or reliance upon, this ISO/IEC publication or any other IEC, ISO or ISO/IEC publications.
- 9) 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.
- 10) Attention is drawn to the possibility that some of the elements of this International Standard may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 29341-17-11 was prepared by UPnP Forum Steering committee¹, was adopted, under the fast track procedure, by subcommittee 25: Interconnection of information technology equipment, of ISO/IEC joint technical committee 1: Information technology.

The list of all currently available parts of the ISO/IEC 29341 series, under the general title *Information technology – UPnP device architecture*, can be found on the IEC web site.

This International Standard has been approved by vote of the member bodies, and the voting results may be obtained from the address given on the second title page.

¹ UPnP Forum Steering committee, UPnP Forum, 3855 SW 153rd Drive, Beaverton, Oregon 97006 USA. See also "Introduction".

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 publication using a colour printer.

1 Overview and Scope

This service definition is compliant with the UPnP Device Architecture version 1.0.

This service-type enables modeling of 'Quality of Service Manager' function capabilities. QoSManager functionality is the combination of QoSManager service and control point functionality that discovers and controls QoSDevice and QoSPolicyHolder services running on the network. The QoSManager function is responsible for requesting, updating, releasing and in general controlling the Quality of Service assigned by networking devices to various traffic streams. The QoSManager service is invoked from an UPnP Control Point to perform the functions related to setting up QoS² for that traffic stream. Once the network is configured with respect to the QoS for the upcoming traffic stream, the QoSManager service will hand back control to the Control Point. This service provides a mechanism for a Control Point to:

- Be agnostic of the QoS capabilities and associated details about the various QoS Devices on the network at the expense of potential network inefficiency.
- Abstract the tasks of setting up, modifying and revoking the QoS associated with every traffic stream
- QoSManager:3 defines new capabilities to set up Parameterized QoS. Control Points may now request reservations for specific resources for the exclusive use of individual streams. Successful resource reservation ensures that streams will always have the bandwidth and other resources to transport time critical data.
- QoSManager:3 defines a method for Control Points to request preemption of existing reservations on the UPnP-QoS network if necessary to admit a new stream.
- QoSManager:3 supports a mechanism for providing a list of blocking streams to a Control Point so that it can handle preemption on its own, if desired.

A QoSManager is a dual-role entity that exposes a QoSManager Service to the Control Point while acting as a Control Point for the QoSPolicyHolder Service and QoSDevice Services running on the network. This document describes the components of the UPnP QoSManager Service and the QoS Manager. The QoS Manager provides the Control Point functionality that discovers and controls QoSDevice Services and the QoSPolicyHolder Services running on the network. Additional information concerning the QoS Manager may be found in:

- UPnP QoS Architecture document [QoS Architecture]
- UPnP QoSDevice Service Definition Document [QOS DEVICE]
- UPnP QoSPolicyHolder Service Definition Document [POLICY HOLDER]

1.1 References & Terms

1.1.1 References

Unless explicitly stated otherwise herein, implementation of the mandatory provisions of any standard referenced by this specification shall be mandatory for compliance with this specification. This clause lists the normative references used in this document and includes the tag inside square brackets that is used for each sub reference:

1.1.1.1 Normative References

[XML] – *Extensible Markup Language (XML) 1.0 (Second Edition)*, T. Bray, J.Paoli, C. M. Sperberg McQueen, E Maler, eds. W3C Recommendations, 6 October 2000.

[DEVICE] - UPnP Device Architecture, version 1.0, UPnP Forum, July 20, 2006.
Available at: <http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0-20060720.pdf>

² Quality of Service

Latest version available at: <http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf>

[POLICY HOLDER] – UPnP QosPolicyHolder:3 Service Document.
Available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosPolicyHolder-v3-Service-20081130.pdf>

Latest version available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosPolicyHolder-v3-Service.pdf>

[QOS DEVICE] – UPnP QosDevice:3 Service Document.
Available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosDevice-v3-Service-20081130.pdf>

Latest version available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosDevice-v3-Service.pdf>

[QDA:3] – UPnP QosDevice:3 Underlying Technology Interface Addendum
Available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosDevice-v3-Addendum-20081130.pdf>

Latest version available at: <http://www.upnp.org/specs/qos/UPnP-qos-QosDevice-v3-Addendum.pdf>

[IANA Portnumbers] – *Port Numbers*, IANA, Available at:
<http://www.iana.org/assignments/port-numbers>

[IANA Protocols] – *Assigned Internet Protocol Numbers*, IANA,
Available at: <http://www.iana.org/assignments/protocol-numbers>

[RFC 2141] – *URN Syntax*, R. Moats, May 1997, Available at:
<http://www.ietf.org/rfc/rfc2141.txt>

[RFC 2211] – *Specification of the Controlled-Load Network Element Service*, J. Wroclawski, September 1997. Available at: <http://www.ietf.org/rfc/rfc2211.txt>

[RFC 2212] – *Specification of Guaranteed Quality of Service*, S. Shenker et al., September 1997. Available at: <http://www.ietf.org/rfc/rfc2212.txt>

[RFC 3339] – *Date and Time on the Internet: Timestamps*, G. Klyne, July 2002. Available at: <http://www.ietf.org/rfc/rfc3339.txt>

[RFC 3927] – *Dynamic Configuration of IPv4 Link-Local Addresses*, Internet Engineering Task Force, S. Cheshire et al., March 2005. Available at: <http://www.ietf.org/rfc/rfc3927.txt>

[RFC 3986] – *Uniform Resource Identifiers (URI): Generic Syntax*, T. Berners-Lee et al., January 2005. Available at: <ftp://ftp.rfc-editor.org/in-notes/rfc3986.txt>

1.1.1.2 Informative References

This clause lists the informative references used in this document and includes the tag inside square brackets that is used for each sub reference:

[CDS:2] – UPnP AV ContentDirectory Service Definition document version 2.0.
Available at: <http://www.upnp.org/specs/av/UPnP-av-ContentDirectory-v2-Service.pdf>
Latest version available at: <http://www.upnp.org/specs/av/UPnP-av-ContentDirectory-v2-Service-20060531.pdf>

[IEEE 802.3-2002] – IEEE Std 802.3, 2000 Edition, Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, LAN MAN Standards Committee of the IEEE Computer Society, 2000.

[IEEE 802.1D] – IEEE Std 802.1D – 2004, Media Access Control (MAC) Bridges, IEEE Computer Society, 2004.

[QoS Architecture] – *UPnP QoS Architecture V3.0* Available at:
<http://www.upnp.org/specs/qos/UPnP-qos-Architecture-v3-20081130.pdf>

Latest version available at: <http://www.upnp.org/specs/qos/UPnP-qos-Architecture-v3.pdf>

[HPAV] – HomePlug AV Specification v1.1, HomePlug Power Alliance, 2007.

[PACKETC] – PacketCable Specification Multimedia Specification,
 Available at: <http://www.packetcable.com/downloads/specs/PKT-SP-MM-I03-051221.pdf>

[RFC 1363] – A Proposed Flow Specification, C. Partridge, September 1992,
 Available at: <http://www.ietf.org/rfc/rfc1363.txt>

1.1.2 Terms and Abbreviations

1.1.2.1 Abbreviations

Table 1-1 — Abbreviations

Abbreviation	Description
AV	Audio/Video
CP	Control Point
DCP	Device Control Protocol
DSCP	Differentiated Services Code Point
IANA	Internet Assigned Numbers Authority
IEEE	Institute of Electrical and Electronics Engineers, Inc.
IETF	Internet Engineering Task Force
IGD	Internet Gateway Device
IP	Internet Protocol
LAN	Local Area Network
MAC	Medium Access Control
PVR	Personal Video Recorder
QD	<u>QoSDevice</u> Service
QM	<u>QoSManager</u> Service
QoS	Quality of Service
QPH	<u>QoSPolicyHolder</u> Service
SRP	Stream Reservation Protocol
STA	<u>Station</u>
STB	Settop Box
TSPEC	Traffic Specification
WMM	Wireless Multi Media
WMM-SA	Wireless Multi Media Scheduled Access
XML	eXtensible Markup Language

1.1.2.2 Terms

1.1.2.2.1 Admission Control

Admission control is the overall procedure that consists of

- Breaking down (by a [QoS Manager](#)) of the end-to-end requirements into per QoS *segment* requirements;
- Requesting the reservation of resources (at the [QoSDevice](#) Service) and the process that is carried out by a [QoSDevice](#) Service to reserve those resources;
- Applying the results from the [QoSPolicyHolder](#) Service to determine the priority and / or importance of the stream and whether it should be admitted at the possible expense of other streams.

The overall result is a yes/no decision of whether the traffic stream can be supported.

1.1.2.2.2 Admission Mechanism

Admission mechanism is the implementation of the resource management scheme of a [QoSDevice](#) Service. The [QoSDevice](#) Service abstracts the properties of a particular technology and the UPnP-QoS interface provides access to the *Admission Mechanism* via the defined actions. The resource management scheme may operate at any layer of the ISO stack, although it frequently uses L2 signaling.

1.1.2.2.3 Admission Policy

Admission policy is the set of rules that is applied in an *admission control* procedure. The rules are applied on the parameters from the [TrafficPolicy](#) as supplied by the [QoSPolicyHolder](#) Service and the Traffic Specification.

1.1.2.2.4 AV Control Point

A UPnP *Control Point* that is used in an AV scenario and utilizes UPnP AV services is called an *AV Control Point*.

1.1.2.2.5 Best Effort

Best effort is a service of the Internet Protocol; i.e. it makes almost no guarantees about a packet other than attempting to transmit it. At the destination, the packet may arrive damaged, it may be out of order (compared to other packets sent between the same hosts), it may be duplicated, or it may be dropped entirely. If reliability is needed, it is provided by upper level protocols transported using IP.

With the introduction of IEEE 802.1D *priority tagging* [IEEE 802.1D] the priority associated with untagged packets is also called "Best Effort".

1.1.2.2.6 Blocked Segments

A *segment* is called *blocked* for the setup of a new traffic stream, if the *admission* on the segment failed due to insufficient *resources* somewhere in the segment.

1.1.2.2.7 Blocking Traffic Stream

An *admitted traffic stream* is called *blocking* with respect to a new traffic stream S, if it uses network resources where the admission of S fails.

1.1.2.2.7.1 Bridge

A *device* that interconnects two or more LANs that use the same Data Link layer protocols above the MAC sublayer, but can use different MAC protocols.

1.1.2.2.8 Complex Type

A *complex type* is defined in the XML Schema Definition Language [XML] .

1.1.2.2.9 Container

A *container* is a UPnP-QoS construct that contains *containers* and/or *fields*. In XML it is expressed through elements of *complex type*.

1.1.2.2.10 Device

A *device* is a functional unit for a particular purpose. A device is not necessarily a *UPnP Device*.

1.1.2.2.11 Element

An *element* is defined in the XML Schema Definition Language [XML] .

1.1.2.2.12 End-to-End Delay

The End-to-End delay of an application payload is defined as the time difference between the delivery of the payload for sending by the application on the source device, and the delivery of the payload to the destination device's application. The time reference used at the source and destination is the same and is known at the application layer. The End-to-End delay experienced by the packets of a stream is not constant and the delay will have a maximum and a minimum, for a period of observation. An application may, for example, require that the maximum End-to-End delay be less than or equal to an actual value it desires. See [QOS DEVICE] for more details.

1.1.2.2.13 End-to-End Jitter

End-to-End Jitter is defined as the difference between the maximum of End-to-End Delay and the minimum of End-to-End Delay.

1.1.2.2.14 Existing User

When pre-empting an existing traffic stream, the user(s) enjoying this stream is/are affected. In a pre-emption discussion, this user is called the "*Existing User*".

1.1.2.2.15 Field

A *field* is a UPnP-QoS construct that provides information on properties, capabilities, etc. In XML it is expressed through an element of simple type.

1.1.2.2.16 Hub

A device, with two or more physical ports, that forwards all traffic received on any individual port to all other ports. This device is also referred to as a "repeater". In a home network, a hub connects networks segments of the same physical medium.

1.1.2.2.17 Interface

The point of interconnection between a device and a network.

1.1.2.2.18 Intermediate Device

An *intermediate device* is physically connected between the Source and Sink device traffic stream. There may be zero or more intermediate device between the Source to Sink.

1.1.2.2.19 Internet Gateway Device (IGD)

Internet Gateway Device is a border device that physically connects the Home Network with a Wide Area Network. This device performs routing. QoS mechanisms associated with routing are not addressed by this architecture; the IGD may present a [*QosDevice*](#) Service on the Home Network.

1.1.2.2.20 L2 Technology

The term *L2 Technology* is short hand to refer to MAC / PHYs of various LAN technologies. Examples include: IEEE 802.3, MOCA, HomePlug.

1.1.2.2.21 Layer 2 Stream Id

A Layer 2 construct to uniquely identify a traffic stream within a *QoS segment*.

1.1.2.2.22 Lease Time

The *lease time* is the finite duration for which the resources are reserved for a traffic stream.

1.1.2.2.23 Link

A *link* is an (optionally bidirectional) direct connection between two devices for data exchange. This is called a link segment in the terminology of [IEEE 802.3-2002] .

1.1.2.2.24 Path

A *path* is the physical course that traffic stream will follow from source to sink. For UPnP-QoS, a path must reside within a single IP subnet, but a path may comprise multiple segments. This is called a link in the terminology of [IEEE 802.3-2002] .

1.1.2.2.25 Path Determination

Path determination is the process of determining the ordered set of [*QosDevice*](#) Service(s) on the path of a given traffic stream.

1.1.2.2.26 Packet Priority

The *Packet Priority* is a Layer 2 parameter that indicates the priority handling requested by the source. Typically this parameter is part of a packet header and indicates the relative importance of the packet compared to other packets. It is used to differentiate packets to determine which are given preferential access to the communication medium. This parameter is typically mapped to an Access Priority value supported by the network device.

1.1.2.2.27 Parameterized QoS

Parameterized QoS refers to a general methodology for obtaining QoS. A contract is made with the network using a set of parameters that define the application's traffic requirements. The parameters typically include bandwidth and delay.

1.1.2.2.28 Policing

Policing is the process of ensuring that no more than the reserved resources are used.

1.1.2.2.29 Policy

Policy is a set of rules that is applied for the allocation of resources.

1.1.2.2.30 Policy Management

Policy Management is a function that applies policies to traffic streams.

1.1.2.2.31 Preempted Traffic Stream

A *preempted traffic stream* is a blocking traffic stream that has its reservation modified or deleted by the [QoS Manager](#).

1.1.2.2.32 Preemption

When an admission request fails due to the lack of resources, the [QoS Manager](#) may attempt to free resources necessary to allow the new traffic stream to reserve them. The process of taking resources from existing admitted traffic streams is called *preemption*.

1.1.2.2.33 Preemption Candidate

A *preemption candidate* is a blocking traffic stream that has been selected for possible preemption by the [QoS Manager](#).

1.1.2.2.34 Preemption Style

A *preemption style* is a UPnP-QoS defined method of how the [QoS Manager](#) selects preemption candidates.

1.1.2.2.35 Prioritized QoS

Prioritized QoS refer to a general methodology for providing QoS by differentiating traffic. Messages types are grouped by order of importance and assigned a priority. Message types assigned a higher priority are given preferential access to the communications medium.

1.1.2.2.36 Quality of Service (QoS)

The term *Quality of Service* (abbreviated QoS) refers to a broad collection of networking capabilities and techniques. The goal of QoS is to improve the user experience of a network's ability to deliver predictable results for sensitive applications such as audio, video, and voice applications. Elements of network performance within the scope of QoS often include bandwidth (throughput), latency (delay), and error rate. There are two broad classes of QoS: data reliability and temporal reliability. Each makes different demands on network technologies. This architecture is primarily concerned with delivering temporal reliability.

1.1.2.2.37 QoS segment

A *QoS segment* is defined as a collection of *interfaces* of [QoSDevice](#) services reporting the same [QoSSegmentId](#) (as defined in [QOS DEVICE]), and the network resources (e.g., wires) through which these interfaces are connected.

1.1.2.2.38 QoS Segment Delay

[QoS Segment Delay](#) represents one of the following: 1. The delay at the [QoSDevice](#) including the interfaces; or 2. The delay introduced by all or part of the QoS Segment between the [QoSDevices](#) including the interfaces. The sum of the [QoS Segment Delay](#) values for [QoSDevices](#) in the path on a QoS segment must be the total delay introduced on that QoS segment. The [QoS Segment Delay](#) for a [QoSDevice](#) service is technology and implementation specific. Details on calculating the QoS Segment Delay can be found in the [QoSDevice](#) Service document and the technology specific addendum [QDA:3].

1.1.2.2.39 QoS Segment Jitter

The QoS Segment Jitter is defined as the difference between the maximum of QoS Segment Delay and the minimum of QoS Segment Delay of the same QoS Segment.

1.1.2.2.40 Resource

In the context of QoS, a *resource* is anything managed in order to support a QoS function. A distinction can be made between device resources and network resources.

1.1.2.2.41 Simple Type

A *simple type* is defined in the XML Schema Definition Language.

1.1.2.2.42 Sink Device

A *sink device* is the receiving end of a traffic stream regardless of other functions on the device.

1.1.2.2.43 Source Device

A *source device* is the originating end of a traffic stream regardless of other functions on the device.

1.1.2.2.44 Subnet

Subnet as defined in [RFC 3927] .

1.1.2.2.45 Switch

A *switch* is a device, with two or more similar or dissimilar physical ports, that sends traffic received on any individual port to the appropriate other port(s) based on layer 2 information.

1.1.2.2.46 Traffic Class

The *Traffic Class* indicates the kind of traffic in the traffic stream. The Traffic Class is used to distinguish, for example, audio from video. The distinction is at the application layer and the Traffic Class is mapped into the applicable layer 2 representations for the L2 Technology bearing the stream. An example is the mapping in IEEE 802.1D, Annex G [IEEE 802.1D].

1.1.2.2.47 Traffic Descriptor

The *TrafficDescriptor* is the representation of a traffic stream for purposes of providing QoS. It is composed of a *TrafficId* and other pertinent information that must be provided by the Control Point.

1.1.2.2.48 Traffic Identifier

A *Traffic Identifier* (*TrafficId*) is a set of information that uniquely identifies a data packet as belonging to a traffic stream. This information is typically used by a packet classifier function to associate a Traffic Specification's QoS contract to the service provided to the Traffic stream. Other technologies may refer to this as a Filter Spec (RFC2205) or Traffic Classifier (IEEE 802.11e).

1.1.2.2.49 Traffic Specification (TSPEC)

A *Traffic Specification (TSPEC)* contains a set of parameters that define the characteristics of the traffic stream. The TSPEC defines the resource requirements for carrying the traffic stream.

1.1.2.2.50 Traffic Stream

A *Traffic Stream* is a unidirectional flow of data. It originates at a source device and terminates at a sink device.

1.1.2.2.51 UPnP Device

A *UPnP Device* is defined in the UPnP Device Architecture[DEVICE] .

2 Service Modeling Definitions

2.1 ServiceType

The following service type identifies a service that is compliant with this template:

urn:schemas-upnp-org:service:QosManager:3

2.2 State Variables

Reader Note: *For first-time reader, it may be more insightful to read the theory of operations first and then the action definitions before reading the state variable definitions.*

2.2.1 XML Fragments as UPnP Arguments

UPnP-QoS often uses XML Fragments as arguments in UPnP actions. The containing UPnP data type is a string. This places restrictions on a string's content; it has to represent a well-formed XML fragment (this includes a complete XML document).

An XML fragment, in adherence to the UPnP V1.0 architecture [DEVICE], MUST be escaped by using the normal XML rules, [XML] Clause 2.4 Character Data and Markup, before embedding it in a SOAP request / response message or an event notification message. The XML escaping rules are summarized:

- The (<) character is encoded as (<)
- The (>) character is encoded as (>)
- The (&) character is encoded as (&)
- The (") character is encoded as (")
- The (') character is encoded as (')

In their XML fragments, implementations MAY use an explicit reference to appropriate namespaces.

Table 2-1 — State Variables

Variable Name	Req. or Opt. ^a	Data Type	Allowed Value ^b	Default Value ^b	Eng. Units
A_ARG_TYPE_TrafficDescriptor	R	String (XML fragment)	See 2.2.2	n/a	n/a
A_ARG_TYPE_TrafficHandle	R	String (XML fragment)	See 2.2.3	n/a	n/a
A_ARG_TYPE_NumTrafficDescriptors	R	ui4	See 2.2.4	n/a	n/a
A_ARG_TYPE_NumPolicyHolders	R	ui4	See 2.2.5	n/a	n/a
A_ARG_TYPE_ListOfTrafficDescriptors	R	String (XML fragment)	See 2.2.6	n/a	n/a
A_ARG_TYPE_QmCapabilities	R	String (XML fragment)	See 2.2.7	n/a	n/a
A_ARG_TYPE_ExtendedTrafficQosInfo	O	String (XML fragment)	See 2.2.8	n/a	n/a
A_ARG_TYPE_ResultedQosType	R	ui4	See 2.2.9	n/a	n/a
^a R = Required, O = Optional, X = Non-standard ^b Values listed in this column are required. To specify standard optional values or to delegate assignment of values to the vendor, you MUST reference a specific instance of an appropriate table below.					

2.2.2 A_ARG_TYPE_TrafficDescriptor

The A_ARG_TYPE_TrafficDescriptor state variable contains information about a particular traffic stream.

2.2.2.1 XML Schema Definition

This is a string containing an XML fragment. It contains information describing a traffic descriptor. The XML fragment in this argument MUST validate against the XML schema for TrafficDescriptor in the XML namespace "http://www.upnp.org/schemas/TrafficDescriptorv1.xsd" which is located at "http://www.upnp.org/schemas/qos/TrafficDescriptor-v3.xsd".

2.2.2.2 Description of fields in the TrafficDescriptor structure

2.2.2.2.1 TrafficHandle

This is an OPTIONAL field in the TrafficDescriptor and is not related to A_ARG_TYPE_TrafficHandle. TrafficHandle is a unique identifier associated with a particular instance of TrafficDescriptor; i.e. a particular traffic stream. It is a unique string generated by the QosManager Service and provided to the Control Point in response to the QM:RequestTrafficQos() or QM:RequestExtendedTrafficQos() action. The QoS Manager must ensure that the generated Traffic Handle string is unique for all Traffic Descriptors on the network. In all subsequent communication between the Control Point and the QosManager Service, TrafficHandle is used to uniquely reference a particular traffic stream.

2.2.2.2.2 TrafficId

This is a MANDATORY structure. TrafficId contains information about identifying/classifying packets that belong to that particular traffic stream. It consists of an XML structure consisting of: SourceAddress, DestinationAddress, SourcePort, DestinationPort, and IpProtocol. The embedded v3TrafficId further contains SourceUuid and DestinationUuid.

2.2.2.2.2.1 SourceAddress

This is an OPTIONAL or MANDATORY field depending on the action. The SourceAddress identifies the source of the traffic stream. A value of 0.0.0.0 represents a 'Don't Care' i.e., the

condition where a Control Point wants to set up QoS for all traffic streams irrespective of the [SourceAddress](#).

2.2.2.2.2.2 [DestinationAddress](#)

This is an OPTIONAL or MANDATORY field depending on the action. The [DestinationAddress](#) identifies the destination of the traffic stream. A value of 0.0.0.0 represents a 'Don't Care' i.e., the condition where a Control Point wants to set up QoS for all traffic streams irrespective of the [DestinationAddress](#).

2.2.2.2.2.3 [SourcePort](#)

This is an OPTIONAL or MANDATORY field depending on the action. The [SourcePort](#) identifies the port of the source of the traffic stream. A value of 0 represents a 'Don't Care' i.e., the condition where a Control Point wants to set up QoS for all traffic streams irrespective of the [SourcePort](#).

2.2.2.2.2.4 [DestinationPort](#)

This is an OPTIONAL or MANDATORY field depending on the action. The [SourcePort](#) identifies the port of the destination of the traffic stream. A value of 0 represents a 'Don't Care' i.e., the condition where a Control Point wants to set up QoS for all traffic streams irrespective of the [DestinationPort](#).

2.2.2.2.2.5 [IpProtocol](#)

This is an OPTIONAL or MANDATORY field depending on the action. The value of [IpProtocol](#) is an IANA assigned IP protocol number from 0-255 (for more information: [IANA Protocols]). A value of 0 represents a 'Don't Care' i.e., the condition where a Control Point wants to set up QoS for all traffic streams irrespective of the [IpProtocol](#).

2.2.2.2.2.6 [v2TrafficId](#)

This is an OPTIONAL structure. It MAY contain a [v3TrafficId](#) structure.

2.2.2.2.2.6.1 [v3TrafficId](#)

This is an OPTIONAL structure under the [TrafficId](#) structure. The [v3TrafficId](#) structure contains two fields: [SourceUuid](#) and [DestinationUuid](#).

2.2.2.2.2.6.1.1 [SourceUuid](#)

This is an OPTIONAL field. The [SourceUuid](#) field contains the UDN of the UPnP Device at the source.

```
UUID ::= (* The UDN of the device including the prefix 'uuid:'.
          Refer to Clause 2.1 of [DEVICE] *)
```

Example:

```
uuid:2fac1234-31f8-11b4-a222-08002b34c003
```

2.2.2.2.2.6.1.2 [DestinationUuid](#)

This is an OPTIONAL field. The [DestinationUuid](#) field contains the UDN of the UPnP Device at the destination.

```
UUID ::= (* The UDN of the device including the prefix 'uuid:'.
          Refer to Clause 2.1 of [DEVICE] *)
```

Example:

uuid:1fac1234-31f8-11b4-a222-08002b34c002

2.2.2.2.3 AvailableOrderedTspecList

This is a MANDATORY structure. AvailableOrderedTspecList contains one or more Tspec components. Their value of TspecIndex reflects the order of preference. The values of TspecIndex MUST be unique within an AvailableOrderedTspecList. TSPECs with smaller TspecIndex values are more preferred. TspecIndex values need not be consecutive numbers. The order of Tspec structures in the AvailableOrderedTspecList is not relevant.

2.2.2.2.3.1 Tspec

This is a MANDATORY structure. Tspec contains a TSPEC (TrafficSpecification) which is a description of the QoS Requirements.

In a UPnP-AV scenario, this information is extracted from the ContentDirectory Service of the MediaServer Device. In the ContentDirectory Service, Tspec is represented either as a string, containing an escaped XML structure, or as an URI pointing to the escaped XML structure. The UPnP-AV Control Point uses CDS:Browse() and/or CDS:Search() action³ calls to acquire the Tspec structure(s) associated with the content.

The XML structure contains the following elements:

- TspecIndex: This is a MANDATORY field. It is a unique numerical index associated with a particular Tspec. The value of TspecIndex indicates preference (as defined by the application or the end user). A Tspec with a smaller index is more preferred compared to a Tspec with a larger index.
- AvTransportUri: This is an OPTIONAL field. It is a string that contains the URI associated with the UPnP-AV content item for which QoS is being requested.
- AvTransportInstanceid: This is an OPTIONAL field. It is an integer representing the unique InstanceID associated with the UPnP AVTransport Service associated with the content item for which QoS is being requested.
- TrafficClass: This is an OPTIONAL field. It contains the traffic class associated with the traffic stream. This is an enumerated variable that can be assigned to one of the following list of values:
 - NetworkControl
 - StreamingControl
 - Voice
 - AV
 - Data
 - Audio
 - Image
 - Gaming
 - Other
 - Background
- v2TrafficSpecification: This is an OPTIONAL structure. It MAY contain a v3TrafficSpecification structure. The v3TrafficSpecification structure contains the parameters as specified in Clause 2.2.2.3.

³ CDS = ContentDirectory Service

2.2.2.2.4 ActiveTspecIndex

This is an OPTIONAL field when provided to the QosManager Service or QosPolicyHolder Service and a MANDATORY field when provided to the QosDevice Service. ActiveTspecIndex contains an integer which indicates the index of the current active Tspec from the AvailableOrderedTspecList. Tspec and AvailableOrderedTspecList are defined above.

2.2.2.2.5 TrafficImportanceNumber

This is an OPTIONAL field when provided to the QosManager Service or QosPolicyHolder Service and a MANDATORY field when provided to the QosDevice Service for Prioritized and Hybrid QoS requests. A TrafficImportanceNumber is an integer with values in the range of 0 through 7. This value follows the numbering scheme for traffic classes as described in IEEE 802.1D Annex G [IEEE 802.1D] and with additional traffic classes described in clause 2.4.1.4.2. This value is used by QosDevice service(s) in the traffic's path to indicate what priority level to utilize when priority tagging the traffic's network packets.

2.2.2.2.6 QosBoundarySourceAddress

This is an OPTIONAL field. If a traffic stream originates outside the UPnP-QoS network's subnet (e.g. on the Internet), the QosBoundarySourceAddress will be treated as the QoS termination point for UPnP-QoS. This parameter is OPTIONAL, because it is applicable only to traffic streams originating outside the home network. It should be noted that this address is not part of the traffic identifier, because the IP packets will carry the IP address of the actual source address. QosBoundarySourceAddress is used by the QoS Manager for decisions related to path determination and device selection.

The "Don't care" situation that may apply for SourceAddress and DestinationAddress IP address does not apply here.

2.2.2.2.7 QosBoundaryDestinationAddress

This is an OPTIONAL field. If a traffic stream terminates outside the UPnP-QoS network's subnet (e.g. on the Internet), the QosBoundaryDestinationAddress will be treated as the QoS termination point for UPnP-QoS. This parameter is OPTIONAL, because it is applicable only to traffic streams terminating outside the home network. It should be noted that this address is not part of the traffic identifier, because the IP packets will carry the IP address of the actual destination address. QosBoundaryDestinationAddress is used by QoS Manager for decisions related to path determination and device selection.

The "Don't care" situation that may apply for SourceAddress and DestinationAddress IP address does not apply here.

2.2.2.2.8 MediaServerConnectionId

This is an OPTIONAL field. This field may be useful in case of UPnP-AV based streaming with multiple Tspec structures. The MediaServerConnectionId may be used by the QosManager Service to identify the traffic stream that is being setup by the UPnP-AV Control Point which is requesting QoS. Please refer to UPnP-QoS Architecture document for more details [QoS Architecture]. The MediaServerConnectionId is obtained from the MediaServer Device via the CM:PrepareForConnection() action⁴.

2.2.2.2.9 MediaRendererConnectionId

This is an OPTIONAL field. This field may be useful in case of UPnP-AV based streaming with multiple Tspec structures. MediaRendererConnectionId may be used by the QosManager Service to identify the traffic stream that is being setup by the UPnP-AV Control Point which

⁴ CM = UPnP-AV ConnectionManager Service

is requesting QoS. Please refer to UPnP-QoS Architecture document for more details [QoS Architecture]. The [MediaRendererConnectionId](#) is obtained from the [MediaRenderer](#) Device via the [CM:PrepareForConnection\(\)](#) action.

2.2.2.2.10 [TrafficLeaseTime](#)

This is an OPTIONAL field. This field contains the lease time associated with a particular traffic stream. The lease time is expressed in seconds. When not specified (which MUST only occur for prioritized streams), it indicates an indefinite lease, that is the [TrafficDescriptor](#) will remain active until it is explicitly removed or the [QosDevice](#) is rebooted. When the [QosDevice](#) Service or [QosManager](#) Service reports the lease time to a Control Point, this is the remaining lease time.

2.2.2.2.11 [v2](#)

This is an OPTIONAL structure. It contains elements that are defined in UPnP-QoS v2 as specified below.

2.2.2.2.11.1 [PolicyHolderId](#)

This is an OPTIONAL field. A [PolicyHolderId](#) is a value that uniquely identifies [QosPolicyHolder](#) Service on the home network. This value informs the [QosManager](#) Service which [QosPolicyHolder](#) Service to query for a [TrafficPolicy](#). The value must be of the following form as described using EBNF:

```
PolicyHolderId    ::= UUID ':' serviceId
UUID              ::= (* The UDN of the QosPolicyHolder service including the
                        prefix 'uuid:'. Refer to Clause 2.1 of [DEVICE] *)
serviceId         ::= (* The serviceId of the QosPolicyHolder service
                        including the prefix 'urn:upnp-org:serviceId:'. Refer
                        to Clause 2.1 of [DEVICE] *)
```

For example:

```
uuid:2fac1234-31f8-11b4-a222-08002b34c003:urn:upnp-org:serviceId:QosPolicyHolder-3a
```

2.2.2.2.11.2 [PolicyLastModified](#)

This is an OPTIONAL field. [PolicyLastModified](#) is an RFC3339 compliant string [RFC 3339] that identifies the wall clock date/time when the policy on a device containing the [QosPolicyHolder](#) Service was last modified. It is recognized that device clocks may not be synchronized. This field is valuable for implementations to provide diagnostic information to the end user.

2.2.2.2.11.3 [PolicyModifyingUserName](#)

This is an OPTIONAL field. [PolicyModifyingUserName](#) is a string that identifies the user who last changed the policy. This field MUST be <= 64 UTF-8 characters. At present this specification does not define the semantics for this field. Examples include, "Dad", "Jimmy", etc. This field is valuable for implementations to provide diagnostic information to the end user.

2.2.2.2.11.4 [PolicyHolderConfigUrl](#)

This is an OPTIONAL field. [PolicyHolderConfigUrl](#) is a string containing a URL on the device that provides the [QosPolicyHolder](#) Service. If the device provides the URL, containing "HTTP" or "HTTPS" as the <protocol>, for its configuration, then the control point may retrieve a page using the provided URL to allow a user to configure the [QosPolicyHolder](#) Service (the configuration method is out-of-scope). The host portion of the URL must be in dotted-decimal IP address form, e.g. "http://10.0.0.131/ config_policy.html".

PolicyHolderConfigURL ::=
Protocol '://' Host [':' Port] [Path ['?' Query]]

2.2.2.2.11.5 v3

This is an OPTIONAL structure. It contains elements that are defined in UPnP-QoS v3 as specified below.

2.2.2.2.11.5.1 QosBoundarySourceUuid

This is an OPTIONAL field. It contains the UDN of the UPnP Device at the QoS Boundary Source.

2.2.2.2.11.5.2 QosBoundaryDestinationUuid

This is an OPTIONAL field. It contains the UDN of the UPnP Device at the QoS Boundary Destination.

2.2.2.2.11.5.3 Critical

This is an OPTIONAL field. The Critical field contains a Boolean that expresses the control point's belief on whether this traffic stream is critical. Streams for entertainment are not critical traffic streams. Examples of critical traffic streams are emergency alerts, emergency calls, etc.

2.2.2.2.11.5.4 PolicyHolderConsultedId

This is an OPTIONAL field. The PolicyHolderConsultedId is a string that contains a value that uniquely identifies the QosPolicyHolder Service that was consulted in setting up QoS for the stream. The format is the same as the format of the PolicyHolderId field.

2.2.2.2.11.5.5 PolicyHolderConsultedType

This is an OPTIONAL field. The value of this integer expresses which version of the behavior to determine which QosPolicyHolder Service to be used was used for this traffic stream. There is no default value.

Table 2-2 — Meaning of PolicyHolderConsultedType parameter

Value	Meaning
0	Default policy
1	There was exactly one <u>QosPolicyHolder</u> Service on the network and that one was used. This is the original UPnP-QoS v1 behavior.
2	The <u>QosPolicyHolder</u> Service that was used is the one indicated by the Control Point. This is the normal UPnP-QoS v2 behavior.
3	The <u>QosPolicyHolder</u> Service that was used is the one preferred by the user.

2.2.2.2.11.5.6 UpdateTrafficLeaseTime

This is an optional boolean field. When a Control Point sets this to “1” in a QoS update request, it indicates that the Control Point is requesting that only the TrafficLeaseTime is updated for the traffic stream.

2.2.2.2.12 OptionalPolicyParameters

This is an OPTIONAL structure. It consists of the fields described in subsequent subclauses below.

2.2.2.2.12.1 UserName

This is an OPTIONAL field. UserName contains a string associated with the user who requested a particular traffic stream. This field identifies the name of the user initiating the UPnP-QoS Action. The field MUST be ≤ 64 UTF-8 characters. At present this specification does not define the semantics for this field. This field represents the name of the user initiating the QoS action. Examples include, “Dad”, “Jimmy”, “CN#1234567”. The latter is an example of a name provided by a service provider to uniquely identify its individual customers.

2.2.2.2.12.2 CpName

This is an OPTIONAL field. CpName contains a string associated with the Control Point requesting QoS for the traffic stream. This field identifies the name of the Control Point initiating the UPnP-QoS Action. The field MUST be ≤ 64 UTF-8 characters. At present this specification does not define the semantics for this field. The CpName may specify the brand name of the Control Point or it may indicate the location of the Control Point such as “living room”.

2.2.2.2.12.3 VendorApplicationName

This is an OPTIONAL field. VendorApplicationName contains a single URI string associated with an application. This field identifies the name of the application initiating the UPnP-QoS Action. For applications specified by vendors, the value of this field MUST begin with “urn:”, followed by an ICANN domain name owned by the vendor, followed by “:application:”, followed by an application name, i.e. the value of this field MUST be constructed as follows:

```
VendorApplicationName ::=
    'urn:' DomainName* ':' Application

DomainName ::= (* Application vendor domain name. This MUST follow the
    syntax specified for Namespace Identifier (NID) in [RFC
    2141]. The domain-name field MUST be ≤ 64 UTF-8
    characters. If the control point is not aware of this
    field, then it MAY be left blank *)

Application ::= 'application:' AppName

AppName ::= (* Name of the application provided by a vendor. This
    must follow the syntax specified for Name Specific
    String (NSS) in [RFC 2141]. The AppName field MUST be
    ≤ 64 UTF-8 characters. The AppName MUST NOT contain a
    colon character. *)
```

For example:

```
urn:example-org:application:MyApp
urn:example-org:application:ScheduledRecording
```

When requesting QoS it may be useful to know the manufacturer and name of the application software client or server that will process the traffic stream.

2.2.2.2.12.4 PortName

This is an OPTIONAL field. PortName contains a single URI **string** associated with the fixed port used by an application for its protocol set-up. This field, if present, shall identify the port number used by the application. The value of this field MUST begin with “urn:”, followed by an ICANN domain name owned by the vendor, followed by “:port:”, followed by the port name. i.e. the value of this field MUST be constructed as follows:

```
PortName ::= 'urn:' DomainName* ':' Port
```

```

DomainName      ::= (* Application vendor domain name. This MUST follow the
                        syntax specified for Namespace Identifier (NID) in [RFC
                        2141]. The domain-name field MUST be <= 64 UTF-8
                        characters. If the control point is not aware of this
                        field, then it MAY be left blank *)

Port            ::= 'port:' PortNumber

PortName        ::= (* An integer value. *)

```

For example:

```
urn:example-org:port:80
```

PortNumber is a fixed port number used by the application for protocol set-up. If the application uses a range of port numbers then the starting port number of that range MUST be specified. This value is either a vendor specific port used for the application or an IANA assigned port number for the application. The list of all the IANA assigned port numbers is maintained by IANA (see [IANA Portnumbers]).

This field indicate the port that is used by an application for its connection set-up. The use of IANA assigned port numbers is RECOMMENDED, but if not available for a particular application, a vendor assigned port number MAY be used.

2.2.2.2.12.5 ServiceProviderServiceName

This is an OPTIONAL field. ServiceProviderServiceName field contains a single URI string associated with a service offered by a service provider. This field identifies the name of the service offered by a service provider. For services provided by service providers, the value of this field MUST begin with “urn:”, followed by an ICANN domain name owned by a service provider, followed by “:service:”, followed by a service name, i.e. the value of this field MUST be constructed as follows:

```

ServiceProviderServiceName
    ::= 'urn:' DomainName* ':' Service

DomainName      ::= (* Application vendor domain name. This MUST follow the
                        syntax specified for Namespace Identifier (NID) in [RFC
                        2141]. The domain-name field MUST be <= 64 UTF-8
                        characters. If the control point is not aware of this
                        field, then it MAY be left blank *)

Service         ::= 'service:' ServName

ServName        ::= (* Name of the service provided by a service vendor.
                        This must follow the syntax specified for Name Specific
                        String (NSS) in [RFC 2141]. The ServName field MUST be
                        <= 64 UTF-8 characters. The ServName MUST NOT contain a
                        colon character. *)

```

For example:

```
urn:example-org:service:VoD
```

2.2.2.3 Description of fields in v3TrafficSpecification structure

This clause defines the parameters of the v3TrafficSpecification structure which is used to characterize the requirements of a specific traffic stream. The traffic specification describes the stream as it flows from Layer 3 to Layer 2. Therefore it includes headers such as the IP header, TCP header, or UDP header but excludes the MAC header, the LLC header, the IEEE 802.1q header, lower-layer FEC bits, or physical layer overhead.

The fields of the v3TrafficSpecification structure are listed in Table 2-3 — Traffic Specification Parameters together with a reference to their normative definition.

Table 2-3 — Traffic Specification Parameters

Name	M/O ^a	Definition	Default
<u>RequestedQosType</u>	O	Clause 2.2.2.3.1	0 (zero)
<u>DataRate</u>	M (if <u>RequestedQosType</u> non-zero)	[RFC 2212] where it is called bucket rate <i>r</i> .	N/A
<u>MinServiceRate</u>	O	Clause 2.2.2.3.7	None
<u>MaxBurstSize</u>	O	[RFC 2212] where it is called bucket depth <i>b</i> .	None
<u>PeakDataRate</u>	O	[RFC 2212] where it is called peak rate <i>p</i> .	Same as <u>DataRate</u> .
<u>ReservedServiceRate</u>	O	[RFC 2212] where it is called Rspec rate term <i>R</i> .	None
<u>TimeUnit</u>	O	Clause 2.2.2.3.4	1000 μsec
<u>MaxPacketSize</u>	O	[RFC 2212] where it is called maximum datagram size <i>M</i> .	1500 octets
<u>E2EMaxDelayHigh</u>	O	Clause 2.2.2.3.11	None
<u>E2EMaxDelayLow</u>	O	Clause 2.2.2.3.13	None
<u>E2EMaxJitter</u>	O	Clause 2.2.2.3.12	None
<u>QosSegmentMaxDelayHigh</u>	O	Clause 2.2.2.3.14	None
<u>QosSegmentMaxDelayLow</u>	O	Clause 2.2.2.3.14.5	None
<u>QosSegmentMaxJitter</u>	O	Clause 2.2.2.3.14.4	None
<u>MaxServiceInterval</u>	O	Clause 2.2.2.3.15	None
<u>MinServiceInterval</u>	O	Clause 2.2.2.3.16	None
<u>LossSensitivity</u>	O	[RFC 1363]	0
<u>ServiceType</u>	O	Clause 2.2.2.3.18	<u>1</u>
^a M = MANDATORY, O = OPTIONAL, X=Non-Standard			

We will introduce the parameters with examples after which the definition follows.

2.2.2.3.1 RequestedQosType

The optional RequestedQosType parameter, if supplied, MUST contain a non-negative integer value. It is used to indicate the type of Quality of Service that is requested with this TrafficDescriptor. The default value is 0.

Table 2-4 — Meaning of RequestedQoSType parameter

Value	Meaning
0	Prioritized QoS If the QoS (update) request succeeds, this traffic stream MUST be successfully set up using prioritized QoS for every QoS segment it traverses. This includes, in particular, prioritized QoS setup for QoS segments that support parameterized QoS.
1	Hybrid QoS If the QoS (update) request succeeds, this traffic stream MUST be successfully admitted using parameterized QoS on every parameterized QoS segment it traverses and MUST be set up using prioritized QoS on every prioritized QoS segments it traverses.
2	Parameterized QoS If the QoS (update) request succeeds, this traffic stream MUST be successfully admitted using parameterized QoS on every parameterized QoS segments it traverses and the traffic stream MUST NOT traverse any non-parameterized QoS segment.
3 and higher	Reserved

2.2.2.3.2 Data rate related parameters

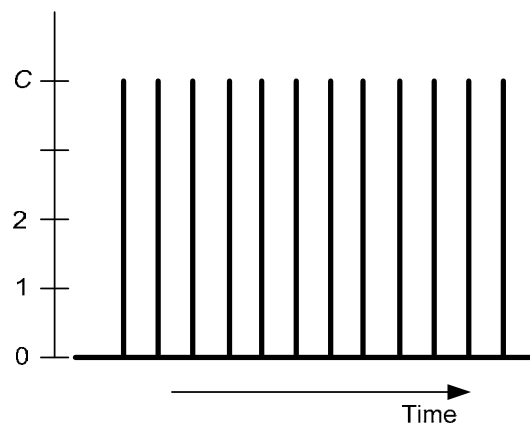
The following subclauses introduce the parameters related to data rate. These clauses are for information only and an attempt to explain the RFCs that contain the normative definitions.

For example, consider a traffic stream that emits $x(t)$ octets at time instant t . For simplicity we ignore propagation delay and therefore when a single packet is available at $t=1$, we have $x(1)$ equal the packet size. The time ranges over the real numbers but the traffic stream is sent in packets and therefore $x(t)=0$ almost everywhere. For ease of notation a stream starts at $t=0$.

2.2.2.3.2.1 Example

A simple example of the traffic generated by a traffic stream is shown in Figure 1. In this case, the amount of data and the frequency it is sent with is constant. Formally, there is a $C > 0$ and a time interval $s > 0$, such that for every positive integer k we have $x(ks) = C$ and for every $t > 0$ that is not an integer multiple of s , we have $x(t) = 0$.

This traffic stream will be modeled by setting the DataRate parameter, that we will describe below, to C .

**Figure 1 — An example traffic stream**

If this traffic stream is served with a rate below C , packets will accumulate in queues. The queues can be emptied after the end of the traffic stream, when no new packets arrive.

If this traffic stream is served at a rate of at least C then packets are not delayed in queues and the total delay is the propagation delay. In Clause 2.2.2.3.11 we describe delay in more detail.

2.2.2.3.3 Data Rate

2.2.2.3.3.1 Definition and Requirements (Normative)

The parameter DataRate is defined in [RFC 2212] where it is called bucket rate r . The parameter DataRate MUST be provided if the value of the RequestedQosType parameter is non-zero. Its unit is octets per second.

2.2.2.3.3.2 Explanation of the Definition (Informative)

The parameter DataRate intends to characterize the long term data rate. To ones skilled in the art it is also called Mean Data Rate or Bucket Rate. For a traffic stream of a finite duration T , the DataRate is defined as

$$\frac{1}{T} \sum_{0 \leq t \leq T} x(t)$$

The unit of DataRate is therefore octets per second.

A crude approximation for a stored file, is obtained by dividing the file length by the duration of the play out of that file.

For traffic streams of infinite duration such as those output from a tuner, the DataRate is defined as

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{0 \leq t \leq T} x(t)$$

For practical purposes a sufficiently large value of T is used to determine the DataRate.

Since serving a traffic stream with a rate below the DataRate will lead to an unbounded accumulation of packets in queues, the traffic stream MUST be served with a rate of at least DataRate.

2.2.2.3.3.3 Example Constant Bit Rate Traffic

In this clause we present an example of using the DataRate for a traffic stream with a traffic corresponding to Figure 2. As can be seen from the figure, the amount of data and the frequency it is sent with is constant. Formally, there is a $C > 0$ and a time interval $s > 0$, such that for every positive integer k we have $x(ks) = C$ and for every $t > 0$ that is not an integer multiple of s , we have $x(t) = 0$.

In this example, the DataRate is C .