Diameter Maintenance and Extensions (DIME)
Internet-Draft
Intended status: Standards Track
Expires: August 30, 2007

J. Korhonen, Ed.
TeliaSonera
H. Tschofenig
Siemens Networks GmbH & Co KG
February 26, 2007

Quality of Service Parameters for RADIUS and Diameter
draft-korhonen-dime-qos-parameters-00.txt

Status of this Memo

Copyright Notice

Abstract

This document defines a number of Quality of Service (QoS) parameters that can be reused for conveying QoS information within RADIUS and Diameter.

The payloads used to carry these QoS parameters are opaque for the AAA client and the AAA server itself and interpreted by the respective Resource Management Function.

Table of Contents

1. Introduction

This document defines a number of Quality of Service (QoS) parameters that can be reused for conveying QoS information within RADIUS and Diameter.

The payloads used to carry these QoS parameters are opaque for the AAA client and the AAA server itself and interpreted by the respective Resource Management Function.

2. Terminology and Abbreviations

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119 [RFC2119].

3. Parameter Overview

3.1. Traffic Model Parameter

The Traffic Model (TMOD) parameter is a container consisting of four sub-parameters:

- o rate (r)
- o bucket size (b)
- o peak rate (p)
- o minimum policed unit (m)

All four sub-parameters MUST be included in the TMOD parameter. The TMOD parameter is a mathematically complete way to describe the traffic source. If, for example, TMOD is set to specify bandwidth only, then set r = peak rate = p, b = large, m = large. As another example if TMOD is set for TCP traffic, then set r = average rate, b = large, p = large.

3.2. Constraints Parameters

<Path Latency>, <Path Jitter>, <Path PLR>, and <Path PER> are QoS parameters describing the desired path latency, path jitter and path bit error rate respectively.

The <Path Latency> parameter refers to the accumulated latency of the packet forwarding process associated with each QoS aware node along the path, where the latency is defined to be the mean packet delay added by each such node. This delay results from speed-of-light propagation delay, from packet processing limitations, or both. The mean delay reflects the variable queuing delay that may be present.

The purpose of this parameter is to provide a minimum path latency for use with services which provide estimates or bounds on additional path delay [RFC2212].

The procedures for collecting path latency information are outside the scope of this document.

The <Path Jitter> parameter refers to the accumulated jitter of the packet forwarding process associated with each QoS aware node along the path, where the jitter is defined to be the nominal jitter added by each such node. IP packet jitter, or delay variation, is defined in Section 3.4 of RFC3393 [RFC3393], (Type-P-One-way-ipdv), and where the selection function includes the packet with minimum delay such that the distribution is equivalent to 2-point delay variation in [Y.1540]. The suggested evaluation interval is 1 minute. This jitter results from packet processing limitations, and includes any variable queuing delay which may be present. The purpose of this parameter is to provide a nominal path jitter for use with services that provide estimates or bounds on additional path delay [RFC2212].

The procedures for collecting path jitter information are outside the scope of this document.

The <Path PLR> parameter refers to the accumulated packet loss rate (PLR) of the packet forwarding process associated with each QoS aware node along the path where the PLR is defined to be the PLR added by each such node.

The <Path PER> parameter refers to the accumulated packet error rate (PER) of the packet forwarding process associated with each QoS aware node, where the PER is defined to be the PER added by each such node.

The <Slack Term> parameter refers to the difference between desired delay and delay obtained by using bandwidth reservation, and which is used to reduce the resource reservation for a flow [RFC2212].

The <Preemption Priority> parameter refers to the priority of the new flow compared with the <Defending Priority> of previously admitted flows. Once a flow is admitted, the preemption priority becomes irrelevant. The <Defending Priority> parameter is used to compare with the preemption priority of new flows. For any specific flow, its preemption priority MUST always be less than or equal to the defending priority. <Admission Priority> and <RPH Priority> provide an essential way to differentiate flows for emergency services, ETS, E911, etc., and assign them a higher admission priority than normal priority flows and best-effort priority flows.

3.3. Traffic Handling Directives

The <Excess Treatment> parameter describes how a QoS aware node will process excess traffic, that is, out-of-profile traffic. Excess traffic MAY be dropped, shaped and/or remarked.

3.4. Traffic Classifiers

Resource reservations might refer to a packet processing with a particular DiffServ per-hop behavior (PHB) [RFC2475] or to a particular QoS class, e.g., Y.1541 QoS class or DiffServ-aware MPLS traffic engineering (DSTE) class type [RFC3564], [RFC4124].

4. Parameter Encoding

4.1. Header

Each QoS parameter is encoded in TLV format using a similar parameter header:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|     Parameter ID      |r|r|r|r|         Length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

M Flag: When set indicates the subsequent parameter MUST be interpreted. Otherwise the parameter can be ignored if not understood.

The r bits are reserved.

Parameter ID: Assigned to each parameter (see below)

4.2. TMOD-1 Parameter

<TMOD-1> = <r> <b> <p> <m> [RFC2210] , [RFC2215]

The above notation means that the 4 <TMOD-1> sub-parameters must be carried in the <TMOD-1> parameter. The coding for the <TMOD-1> parameter is as follows:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |M|r|r|r|           1           |r|r|r|r|         4             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  TMOD Rate-1 [r] (32-bit IEEE floating point number)          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  TMOD Size-1 [b] (32-bit IEEE floating point number)          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Peak Data Rate-1 [p] (32-bit IEEE floating point number)     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Minimum Policed Unit-1 [m] (32-bit unsigned integer)         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The <TMOD> parameters are represented by three floating point numbers in single-precision IEEE floating point format followed by one 32-bit integer in network byte order. The first floating point value is the rate (r), the second floating point value is the bucket size (b), the third floating point is the peak rate (p), and the first unsigned integer is the minimum policed unit (m).

When r, b, and p terms are represented as IEEE floating point values, the sign bit MUST be zero (all values MUST be non-negative). Exponents less than 127 (i.e., 0) are prohibited. Exponents greater than 162 (i.e., positive 35) are discouraged, except for specifying a peak rate of infinity. Infinity is represented with an exponent of all ones (255) and a sign bit and mantissa of all zeroes.

4.3. TMOD-2 Parameter

A description of the semantic of the parameter values can be found in [RFC2215]. The <TMOD-2> parameter may be needed in a DiffServ environment. The coding for the <TMOD-2> parameter is as follows:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |M|r|r|r|           2           |r|r|r|r|         4             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  TMOD Rate-2 [r] (32-bit IEEE floating point number)          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  TMOD Size-2 [b] (32-bit IEEE floating point number)          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Peak Data Rate-2 [p] (32-bit IEEE floating point number)     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |  Minimum Policed Unit-2 [m] (32-bit unsigned integer)         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

When r, b, and p terms are represented as IEEE floating point values, the sign bit MUST be zero (all values MUST be non-negative).

Exponents less than 127 (i.e., 0) are prohibited. Exponents greater than 162 (i.e., positive 35) are discouraged, except for specifying a peak rate of infinity. Infinity is represented with an exponent of all ones (255) and a sign bit and mantissa of all zeroes.

4.4. Path Latency Parameter

A description of the semantic of the parameter values can be found in [RFC2210],[RFC2215]. The coding for the <Path Latency> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           3           |r|r|r|r|         1             |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                Path Latency (32-bit integer)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Path Latency is a single 32-bit integer in network byte order. The composition rule for the <Path Latency> parameter is summation with a clamp of (2**32 - 1) on the maximum value. The latencies are average values reported in units of one microsecond. A system with resolution less than one microsecond MUST set unused digits to zero. The total latency added across all QoS aware nodes along the path can range as high as (2**32)-2.

4.5. Path Jitter Parameter

The coding for the <Path Jitter> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           4           |r|r|r|r|         4             |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          Path Jitter STAT1(variance) (32-bit integer)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Path Jitter STAT2(99.9%-ile) (32-bit integer)        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Path Jitter STAT3(minimum Latency) (32-bit integer)     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Path Jitter STAT4(Reserved)        (32-bit integer)     |
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Path Jitter is a set of four 32-bit integers in network byte order. The Path Jitter parameter is the combination of four statistics describing the Jitter distribution with a clamp of (2**32 - 1) on the maximum of each value. The jitter STATs are reported in units of one microsecond.

4.6. Path PLR Parameter

The coding for the <Path PLR> parameter is as follows:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |M|r|r|r|           5           |r|r|r|r|         1             |
 +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |             Path Packet Loss Ratio (32-bit floating point)    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Path PLR is a single 32-bit single precision IEEE floating point number in network byte order. The PLRs are reported in units of 10^-11. A system with resolution less than one microsecond MUST set unused digits to zero. The total PLR added across all QoS aware nodes can range as high as 10^-1.

4.7. Path PER Parameter

The coding for the <Path PLR> parameter is as follows:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |M|r|r|r|           6           |r|r|r|r|         1             |
 +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
 |             Path Packet Error Ratio (32-bit floating point)   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Path PER is a single 32-bit single precision IEEE floating point number in network byte order. The PERs are reported in units of 10^-11. A system with resolution less than one microsecond MUST set unused digits to zero. The total PER added across all QoS aware nodes can range as high as 10^-1.

4.8. Slack Term> Parameter

A description of the semantic of the parameter values can be found in [RFC2212], [RFC2215]. The coding for the <Path PLR> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           7           |r|r|r|r|         1             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Slack Term [S]  (32-bit integer)                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Slack Term parameter S is nonnegative and is measured in microseconds. S is represented as a 32-bit integer. Its value can range from 0 to (2**32)-1 microseconds.

4.9. Preemption Priority amp; Defending Priority Parameters

A description of the semantic of the parameter values can be found in [RFC3181].

The coding for the <Preemption Priority> & <Defending Priority> sub-parameters is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           8           |r|r|r|r|         1             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Preemption Priority        |      Defending Priority       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Preemption Priority: The priority of the new flow compared with the defending priority of previously admitted flows. Higher values represent higher priority.

Defending Priority: Once a flow is admitted, the preemption priority becomes irrelevant. Instead, its defending priority is used to compare with the preemption priority of new flows.

As specified in [RFC3181], <Preemption Priority> & <Defending Priority> are 16-bit integer values.

4.10. Admission Priority Parameter

A description of the semantic of the parameter values can be found in [Y.1571]. The coding for the <Admission Priority> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           9           |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Admis.Priority|                  (Reserved)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

High priority flows, normal priority flows, and best-effort priority flows can have access to resources depending on their admission priority value as follows:

Admission Priority:

0 - best-effort priority flow
1 - normal priority flow
2 - high priority flow
255 - not used

A reservation without an <Admission Priority> parameter (i.e., set to 255) MUST be treated as a reservation with an <Admission Priority> = 1.

4.11. RPH Priority Parameter

A description of the semantic of the parameter values can be found in [RFC4412]. The coding for the <RPH Priority> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           10          |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         RPH Namespace         | RPH Priority  |   (Reserved)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

[RFC4412] defines a resource priority header (RPH) with parameters "RPH Namespace" and "RPH Priority" combination, and if populated is

applicable only to flows with high admission priority, as follows:

RPH Namespace:

0 - dsn
1 - drsn
2 - q735
3 - ets
4 - wps
255 - not used

Each namespace has a finite list of relative priority-values. Each is listed here in the order of lowest priority to highest priority.

RPH Priority:

4 - q735.4
3 - q735.3
2 - q735.2
1 - q735.1
0 - q735.0

4 - ets.4
3 - ets.3
2 - ets.2
1 - ets.1
0 - ets.0

4 - wps.4
3 - wps.3
2 - wps.2
1 - wps.1
0 - wps.0

For the 4 priority parameters, the following cases are permissible (procedures specified in references):

```
1 parameter: <Admission Priority> [Y.1571]
2 parameters: <Admission Priority>, <RPH Priority> [RFC4412]
2 parameters: <Preemption Priority>, <Defending Priority> [RFC3181]
3 parameters: <Preemption Priority>, <Defending Priority>,
             <Admission Priority> [3GPP-1, 3GPP-2, 3GPP-3]
4 parameters:  <Preemption Priority>, <Defending Priority>,
             <Admission Priority>, <RPH Priority> [3GPP-1, 3GPP-2,
             3GPP-3]
```

It is permissible to have <Admission Priority> without <RPH Priority>, but not permissible to have <RPH Priority> without <Admission Priority> (alternatively <RPH Priority> is ignored in instances without <Admission Priority>).

4.12. Excess Treatment Parameter

The coding for the <Excess Treatment> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           11          |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Excess Trtmnt | Remark Value  |         Reserved                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Excess Treatment: Indicates how the QoS aware node should process out-of-profile traffic, that is, traffic not covered by the <Traffic> parameter. Allowed values are as follows:

- 0: drop
- 1: shape
- 2: remark
- 3: no metering or policing is permitted

The default excess treatment in case that none is specified is that there are no guarantees to excess traffic, i.e., a QoS aware node can do what it finds suitable.

When excess treatment is set to 'drop', all marked traffic MUST be dropped by a QoS aware node.

When excess treatment is set to 'shape', it is expected that the QoS Desired object carries a TMOD parameter. Excess traffic is to be shaped to this TMOD. When the shaping causes unbounded queue growth at the shaper traffic can be dropped.

When excess treatment is set to 'remark', the excess treatment parameter MUST carry the remark value. For example, packets may be remarked to drop remarked to pertain to a particular QoS class. In the latter case, remarking relates to a DiffServ-type model, where packets arrive marked as belonging to a certain QoS class, and when they are identified as excess, they should then be remarked to a different QoS Class.

If 'no metering or policing is permitted' is signaled, the QoS aware node should accept the excess treatment parameter set by the sender with special care so that excess traffic should not cause a problem. To request the Null Meter [RFC3290] is especially strong, and should be used with caution.

4.13. PHB Class Parameter

A description of the semantic of the parameter values can be found in [RFC3140]. The coding for the <PHB Class> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           12          |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| DSCP      |0 0 0 0 0 0 0 0 0 0|            (Reserved)         |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

As prescribed in [RFC3140], the encoding for a single PHB is the recommended DSCP value for that PHB, left-justified in the 16 bit field, with bits 6 through 15 set to zero.

The encoding for a set of PHBs is the numerically smallest of the set of encodings for the various PHBs in the set, with bit 14 set to 1. (Thus for the AF1x PHBs, the encoding is that of the AF11 PHB, with bit 14 set to 1.)

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| DSCP      |0 0 0 0 0 0 0 0 X 0|
+---+---+---+---+---+---+---+---+
```

PHBs not defined by standards action, i.e., experimental or local use PHBs as allowed by [RFC2474]. In this case an arbitrary 12 bit PHB identification code, assigned by the IANA, is placed left-justified in the 16 bit field. Bit 15 is set to 1, and bit 14 is zero for a single PHB or 1 for a set of PHBs. Bits 12 and 13 are zero.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      PHD ID CODE      |0 0 X 0|
+---+---+---+---+---+---+---+---+
```

Bits 12 and 13 are reserved either for expansion of the PHB identification code, or for other use, at some point in the future.

In both cases, when a single PHBID is used to identify a set of PHBs (i.e., bit 14 is set to 1), that set of PHBs MUST constitute a PHB Scheduling Class (i.e., use of PHBs from the set MUST NOT cause intra-microflow traffic reordering when different PHBs from the set are applied to traffic in the same microflow). The set of AF1x PHBs [RFC2597] is an example of a PHB Scheduling Class. Sets of PHBs that do not constitute a PHB Scheduling Class can be identified by using more than one PHBID.

The registries needed to use [RFC3140] already exist. Hence, no new registry needs to be created for this purpose.

4.14. DSTE Class Type Parameter

A description of the semantic of the parameter values can be found in [RFC4124]. The coding for the <DSTE Class Type> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           13          |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|DSTE Cls. Type |                (Reserved)                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

DSTE Class Type: Indicates the DSTE class type. Values currently allowed are 0, 1, 2, 3, 4, 5, 6, 7. A value of 255 (all 1's) means that the <DSTE Class Type> parameter is not used.

4.15. Y.1541 QoS Class Parameter

A description of the semantic of the parameter values can be found in [Y.1541]. The coding for the <Y.1541 QoS Class> parameter is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|M|r|r|r|           14          |r|r|r|r|          1            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Y.1541 QoS Cls.|                (Reserved)                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Y.1541 QoS Class: Indicates the Y.1541 QoS Class. Values currently allowed are 0, 1, 2, 3, 4, 5, 6, 7. A value of 255 (all 1's) means that the <Y.1541 QoS Class> parameter is not used.

Class 0:

Mean delay <= 100 ms, delay variation <= 50 ms, loss ratio <= 10^-3. Real-time, highly interactive applications, sensitive to jitter. Application examples include VoIP, Video Teleconference.

Class 1:

Mean delay <= 400 ms, delay variation <= 50 ms, loss ratio <= 10^-3. Real-time, interactive applications, sensitive to jitter. Application examples include VoIP, Video Teleconference.

Class 2:

Mean delay <= 100 ms, delay variation unspecified, loss ratio <= 10^-3. Highly interactive transaction data. Application examples include signaling.

Class 3:

Mean delay <= 400 ms, delay variation unspecified, loss ratio <= 10^-3. Interactive transaction data. Application examples include signaling.

Class 4:

Mean delay <= 1 sec, delay variation unspecified, loss ratio <= 10^-3. Low Loss Only applications. Application examples include short transactions, bulk data, video streaming.

Class 5:

Mean delay unspecified, delay variation unspecified, loss ratio unspecified. Unspecified applications. Application examples include traditional applications of default IP networks.

Class 6:

Mean delay <= 100 ms, delay variation <= 50 ms, loss ratio <= 10^-5. Applications that are highly sensitive to loss, such as television transport, high-capacity TCP transfers, and TDM circuit emulation.

Class 7:

Mean delay <= 400 ms, delay variation <= 50 ms, loss ratio <= 10^-5. Applications that are highly sensitive to loss, such as television transport, high-capacity TCP transfers, and TDM circuit emulation.

5. IANA Considerations

This document reuses the namespace created in [I-D.ietf-nsis-qspec].

No actions are required by IANA.

6. Security Considerations

This document does not raise any security concerns as it only defines QoS parameters.

7. Acknowledgements

The authors would like to thank the NSIS QSPEC draft authors, the NSIS working group chairs (John Loughney and Martin Stiemerling) and the former and the current Transport Area Directors for their help.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", RFC 2210, September 1997.

[RFC2212] Shenker, S., Partridge, C., and R. Guerin, "Specification of Guaranteed Quality of Service", RFC 2212, September 1997.

[RFC2215] Shenker, S. and J. Wroclawski, "General Characterization Parameters for Integrated Service Network Elements", RFC 2215, September 1997.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998.

[RFC2597] Heinanen, J., Baker, F., Weiss, W., and J. Wroclawski,

"Assured Forwarding PHB Group", RFC 2597, June 1999.

[RFC3140] Black, D., Brim, S., Carpenter, B., and F. Le Faucheur, "Per Hop Behavior Identification Codes", RFC 3140, June 2001.

[RFC3181] Herzog, S., "Signaled Preemption Priority Policy Element", RFC 3181, October 2001.

[RFC3290] Bernet, Y., Blake, S., Grossman, D., and A. Smith, "An Informal Management Model for Diffserv Routers", RFC 3290, May 2002.

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.

[RFC3564] Le Faucheur, F. and W. Lai, "Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering", RFC 3564, July 2003.

[RFC4124] Le Faucheur, F., "Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering", RFC 4124, June 2005.

[RFC4412] Schulzrinne, H. and J. Polk, "Communications Resource Priority for the Session Initiation Protocol (SIP)", RFC 4412, February 2006.

[Y.1541] "Network Performance Objectives for IP-Based Services", , 2006.

8.2. Informative References

[I-D.ietf-nsis-qspec] Ash, J., "QoS NSLP QSPEC Template", draft-ietf-nsis-qspec-15 (work in progress), February 2007.

[Y.1540] "Internet Protocol Data Communication Service - IP Packet Transfer and Availability Performance Parameters", , December 2002.

Authors' Addresses

Jouni Korhonen (editor)
TeliaSonera
Teollisuuskatu 13
Sonera FIN-00051
Finland

Email: jouni.korhonen@teliasonera.com

Hannes Tschofenig
Siemens Networks GmbH & Co KG
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: Hannes.Tschofenig@siemens.com
URI: http://www.tschofenig.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).