

DetNet Working Group  
INTERNET-DRAFT  
Intended Status: Standards Track  
Expires: July 11, 2022

D. Trossen  
Huawei  
F.-J. Goetz  
J. Schmitt  
Siemens  
January 11, 2022

**RSVP for TSN Networks**  
**draft-trossen-detnet-rsvp-tsn-01.txt**

**Abstract**

This document provides a solution for control plane signaling by virtue of proposing changes to RSVP signaling with deterministic services at the underlying TSN enabled layer. The solution covers distributed, centralized, and hybrid signaling scenarios in the TSN and SDN domain. The proposed changes to RSVP IntServ, called RSVP TSN in the remainder of this document, provide a better integration with Layer 2 technologies for resource reservation, for which we outline example API specifications for the realization of RSVP TSN.

**Status of this Memo**

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

**Copyright and License Notice**

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

|                      |  |                    |
|----------------------|--|--------------------|
| <a href="#">1.</a>   | Introduction . . . . .   | <a href="#">3</a>  |
| <a href="#">1.1.</a> | Terminology . . . . .  | <a href="#">4</a>  |
| <a href="#">2.</a>   | Use Cases . . . . .  | <a href="#">4</a>  |
| <a href="#">3.</a>   | Design Rationale . . . . .   | <a href="#">6</a>  |
| <a href="#">3.1.</a> | RSVP IntServ vs RSVP TSN Data Plane Model . . . . .                                  | <a href="#">6</a>  |
| <a href="#">3.2.</a> | RSVP IntServ vs RSVP TSN Resource Reservation Styles . . . . .                       | <a href="#">6</a>  |
| <a href="#">3.3.</a> | RSVP IntServ vs RSVP TSN Object Definitions . . . . .                                | <a href="#">7</a>  |
| <a href="#">3.4.</a> | RSVP IntServ vs RSVP TSN Flow Specification . . . . .                                | <a href="#">7</a>  |
| <a href="#">4.</a>   | RSVP TSN . . . . .   | <a href="#">9</a>  |
| 4.1.                 | Layer Interactions between RSVP TSN and Lower Layer<br>Resource Allocation . . . . . | <a href="#">9</a>  |
| <a href="#">4.2.</a> | API for Deterministic QoS (dQoS) . . . . .   | <a href="#">10</a> |
| <a href="#">4.3.</a> | DnFlow Signaling Interface (DnFSI) . . . . .   | <a href="#">10</a> |
| <a href="#">4.4.</a> | DnFlow Transport Interface (DnFTI) . . . . .   | <a href="#">13</a> |
| <a href="#">4.5.</a> | RSVP TSN Message Formats . . . . .   | <a href="#">14</a> |
| <a href="#">5.</a>   | Security Considerations . . . . .  | <a href="#">14</a> |
| <a href="#">6.</a>   | IANA Considerations . . . . .  | <a href="#">14</a> |
| <a href="#">7.</a>   | Conclusion . . . . .   | <a href="#">14</a> |
| <a href="#">8.</a>   | References . . . . .   | <a href="#">14</a> |
| <a href="#">8.1.</a> | Normative References . . . . .   | <a href="#">14</a> |
| <a href="#">8.2.</a> | Informative References . . . . .   | <a href="#">15</a> |



## 1. Introduction

The authors in [[ID.malis-detnet-controller-plane-framework](#)] provide an overview of the DetNet control plane architecture along three possible classes, namely (i) fully distributed control plane utilizing dynamic signaling protocols, (ii) a centralized, SDN-like, control plane, and (iii) a hybrid control plane.

The Time-Sensitive Networking (TSN) Task Group (TG) is a part of the IEEE 802.1 Working Group (WG) (<https://1.ieee802.org/tsn/>). The charter of the TSN TSG is to provide deterministic services for time sensitive applications through IEEE 802 networks, i.e., guaranteed packet transport with bounded latency, low packet delay variation, and low packet loss.

The TSN TG has developed basic data plane techniques for providing deterministic services within an IEEE 802.1Q network. Key aspects are to provide resource reservation for deterministic services by making use of a separate queue, access control, and determining the upper-, lower- and in-class interference on the egress side for bounded latency. This model for traffic from time sensitive applications, called TSN model, and the associated data plane techniques for time sensitive traffic can be applied to different lower layer network technologies and is not limited to IEEE 802.1Q bridges. DetNet uses for its DnFlows deterministic services provided by the lower layer network technologies.

While many deterministic IP deployments make use of pre-planned flow allocations, using an often centrally managed approach, this draft investigates the use of RSVP [[RFC2205](#)] for allowing endpoints to dynamically signal deterministic IP connectivity in combination with underlying Layer 2 mechanisms, specifically those used for TSN networks. When doing so, considerations arise for the development of a Layer2-specific RSVP protocol, called RSVP TSN in the following.

This document will outline use cases for RSVP TSN, followed by the design rationale and specification for the proposed RSVP TSN protocol.

Note that the document does NOT cover aspects of traffic engineering, specifically for a more detnet-focused revision of RSVP-TE. However, the work in this draft is meant to provide more insights into the possible working of RSVP for detnet (here focused over a specific L2 technology, namely TSN), which may in turn be used for a more general work on detnet-specific extensions needed for RSVP overall. As such, this document has been narrowed in scope from its previous version in [[ID-trossen-detnet-control-signaling](#)].



While this document has originally targeted the DetNet working group by providing a suitable control plane signaling method for deployments with TSN underlays, discussions are ongoing as to where to ultimately place the work in this draft. Possible candidates here are the TEAS (Traffic Engineering Architecture and Signaling) or CCAMP (Common Control and Measurement Plane) WG.

### 1.1. Terminology

This document uses the terminology established in the DetNet Architecture [RFC8655], and the reader is assumed to be familiar with that document and its terminology.

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 RFC 2119 [RFC2119].

## 2. Use Cases

A deterministic network [RFC8655] is composed of DetNet-enabled end systems and DetNet relay nodes which deliver deterministic services. As shown in Figure 1, TSN-enabled end systems can still make use of deterministic networking when they are connected to an DetNet edge node supporting service proxy function to establish a deterministic end-to-end service.

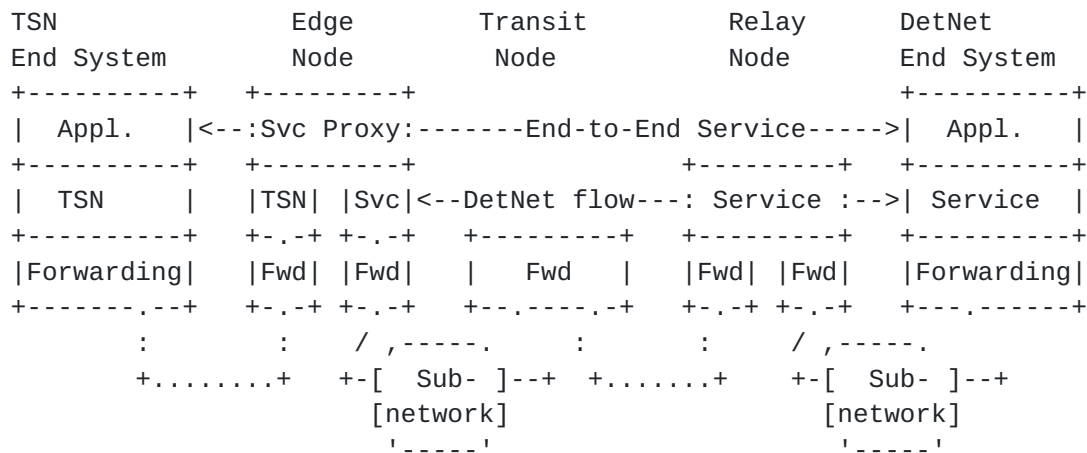


Figure 1 : Deterministic Network with TSN-enabled End Systems

In principle, three use cases are of interest for the establishment of deterministic end-to-end service over deterministic networks:

- (a) DetNet-enabled edge nodes with service proxy on both side because the connected source and destination are TSN-enabled end systems
- (b) Detnet enabled edge nodes with service proxy on one side because



the connected source or destination are TSN-enabled end systems and on the other side the connected source or destination are Detnet-enabled end-systems

- (c) DetNet-enabled end systems are connected to a network supporting end-to-end deterministic services

For the establishment of deterministic end-to-end connectivity based on DnFlows, an end-2-end signaling protocol called RSVP TSN for DnFlows is proposed. To achieve deterministic QoS, access control for a DnFlow is required because each DnFlow must be known by the network supporting DetNet.

The establishment of deterministic end-to-end connectivity is in principle comparable with the establishment of TCP connectivity. The main difference is that all network elements must take active part in the establishment of a deterministic end-to-end connectivity.

RSVP TSN is an option which can be used to signal DnFlow information between

- a) DetNet-enabled edge nodes,
- b) DetNet-enabled edge nodes and DetNet-enabled end system,
- c) DetNet-enabled end systems,
- d) DetNet-enabled end system and first DetNet-enabled relay node,
- e) and DetNet-enabled relay node

NOTE: the next revision of the draft will include a more elaborate reasoning as to why a dynamic signaling may be desirable, possibly alongside managed approaches.

Several years ago, the IETF has introduced RSVP Intserv to exchange flow information for integrated services. Because deterministic service based on TSN differs from integrated services, additional RSVP object definitions are required for RSVP TSN.

Goal of this contribution is to use RSVP TSN for signaling DnFlow information to dynamically establish deterministic end-to-end connectivity. DetNet-enabled end systems support RSVP TSN. There is no need for edge nodes with proxy services. DetNet-unaware or TSN-aware end-systems presume edge nodes supporting proxy services when they want have benefit from DetNet.



In the detnet stack model [[RFC 8938](#)], "Resource allocation" is located in the forwarding sub-layer. In this document, the term "Signaling" is used instead of the term "Resource allocation". One reason for using the term "Signaling" is because the lower layer network technologies like IEEE 802.1Q with TSN enhancements are responsible to allocate queuing, bandwidth and latency resources to provide deterministic services.

### **3. Design Rationale**

IntServ and TSN have defined different models providing deterministic QoS. This section will explore the design rationale behind the development of RSVP TSN. It also outlines aspects derived from the underlying TSN capable lower layer network technology before highlighting key design considerations for the presentation of RSVP TSN in [Section 4](#).

#### **3.1. RSVP IntServ vs RSVP TSN Data Plane Model**

The RSVP IntServ [[RFC2212](#)] model provides a flow bandwidth driven latency model with a separate transmission queue per flow. RSVP IntServ assumes a weighted fair queuing (WFQ) at the data plane, where a listener is able to influence therefore the latency through the reserved bandwidth per flow.

RSVP TSN assumes deterministic services are provided by lower layer network technologies supporting the TSN model. The TSN model itself is in contrasts with the RSVP IntServ [[RFC2212](#)] model. Lower layer network technologies providing deterministic services for traffic from time sensitive applications make use of separate queues, access control, resource reservation and determine the upper-, lower- and in-class interference on the egress side for bounded latency

#### **3.2. RSVP IntServ vs RSVP TSN Resource Reservation Styles**

RSVP IntServ has introduced the notion of 'sessions' to maintain different kinds of deterministic end-to-end connectivity and resource styles, namely fixed (i) filter style, (ii) shared explicit style, and (iii) wildcard filter style - see [[RFC2205](#)]. The receiver controls sender selection and resource styles selection. The receiver is also able to influence latency for a flow by requesting certain amount of bandwidth.

RSVP TSN splits the control over sender selection and resource styles, due to the given TSN data plane model. The resource style is controlled by the sender and the sender selection is controlled by the receiver. The Receiver cannot influence bandwidth for a DnFlow.



The resource style 'coordinated share' is introduced in RSVP TSN to support a large amount of small DnFlows with small data usage. Multiple separate resource reservations on lower layer for small DnFlows could become very inefficient.

| Sender Selection | Resource Style |           |                            |  |
|------------------|----------------|-----------|----------------------------|--|
|                  | Distinct       | Shared    | NEW:<br>Coordinated Shared |  |
| Explicit         | supported      | supported | supported                  |  |
| Wildcard         |                | supported |                            |  |

Figure 2: Resource Style and Sender Selection [[RFC2205](#)]

### 3.3. RSVP IntServ vs RSVP TSN Object Definitions

Due to the differences described above, not all object definitions from RSVP IntServ can be applied to RSVP TSN. Also, not all features are supported in the same way as is done by RSVP IntServ since RSVP TSN assumes a deterministic service to be provided by the lower network layer.

For instance, IEEE 802.1Q networks with TSN enhancements provides deterministic services by a layer 2 protocol for resource allocation for Sstreams [IEEE P802.1Qdd - Resource Allocation Protocol]. Such an allocated Sstream can transport one or multiple DnFlows. A StreamID is used for the identification at the layer 2 control plane.

To correlate DnFlow with their lower layer transport streams a stream identifier information must be distributed by RSVP TSN. This is only one of the reasons for introducing additional RSVP object definitions.

### 3.4 RSVP IntServ vs RSVP TSN Flow Specification

In RSVP IntServ, the flow specification describes both the characteristics of the traffic sent by the source and the service requirements of the application. The flow specification of RSVP IntServ is token bucket based. The sender TSpec is a description of the allowed traffic characteristics for which service is being requested. Each receiver describes by RSpec the service it desires to receive. The RSpec is carried from the receiver to the intermediate network elements and flows upstream towards the sender. It may be used or updated at the intermediate network elements before arriving the sender. The ADSpec object carries information which is generated



at either data sources or intermediate network elements, is flowing downstream towards receivers.

In RSVP TSN, the sender TSpec information is also a description of the allowed traffic characteristics for which service is being requested. The receiver cannot describe the service it desires to receive. The traffic specification itself can be token bucket based but also variants based on intervals are supported. RSVP TSN does not support RSpec. It is not able to support heterogeneous receivers where each makes reservation requests with different QoS requirements on per DnFlow session.

These differences pose a number of questions:

1. Is RSVP IntServ (as defined in [[RFC2212](#)]) the right starting point to deliver DnFlow information and trigger resource allocation on lower layer network technologies supporting the TSN model?
2. How to efficiently map the different reservation styles of RSVP TSN (originally introduced by RSVP IntServ) onto the TSN data plane model?
3. What is the nature of the interface between RSVP TSN and lower layer resource reservation?
4. How does the binding between DnFlow signaling of RSVP TSN and lower layer resource reservation look like?
5. Which of the different RSVP TSN traffic specifications shall be supported?

Note: Different traffic specifications exist for an efficient mapping of traffic specification to scheduling model.

|                          | Time based<br>Scheduling                                  | Token Bucket<br>based<br>Scheduling                         | Priority based<br>(none shaping<br>network nodes) |
|--------------------------|---|---|---|
| Stream/<br>Flow<br>Based | Proposal:<br>Dampers with<br>Forward Traffic<br>isolation | Asynchronous<br>Traffic Shaping<br>(ATS)<br>(IEEE 802.1Qcr) | Highest<br>(static)<br>priority                   |
| Class<br>Based           | Cyclic Queuing &<br>Forwarding (CQF)<br>(IEEE 802.1Qch)   | Credit-Based<br>Shaper (CBSA)<br>(IEEE 802.1Qav)            |   |

Figure 3: Comparison of TSN and RSVP-IntServ Models



The proposal for dumper is discussed within the IEEE 802.1 TSN WG (see <https://www.ieee802.org/1/files/public/docs2020/new-specht-dampers-fti-0620-v02.pdf>).

For instance, the Resource-Allocation-Protocol (RAP) [RAP IEEE] introduces templates to describe traffic class for streams with its scheduling model and the associated traffic specification for streams.

#### 4. RSVP TSN

This section specifies the APIs for RSVP TSN, the message formats, and outlines the layer and node interactions in an RSVP TSN based system.

##### 4.1. Layer Interactions between RSVP TSN and Lower Layer Resource Allocation

Figure 4 provides an overview of the interactions between lower layer resource allocation and DnFlow signaling in a network deployment as an elaboration of the elements in Figure 1, also illustrating the various interfaces described in the following sections.

The application utilizes a generalized API for deterministic QoS (dQoS), which controls and signals the establishment DnFlow via the upper API of RSVP TSN. The latter is called DnFlow-Signaling-Interface(uRSVPDnFSI) in this contribution.

DetNet end nodes utilize RSVP TSN to distribute DnFlow information by end-to end signaling over DetNet Route.

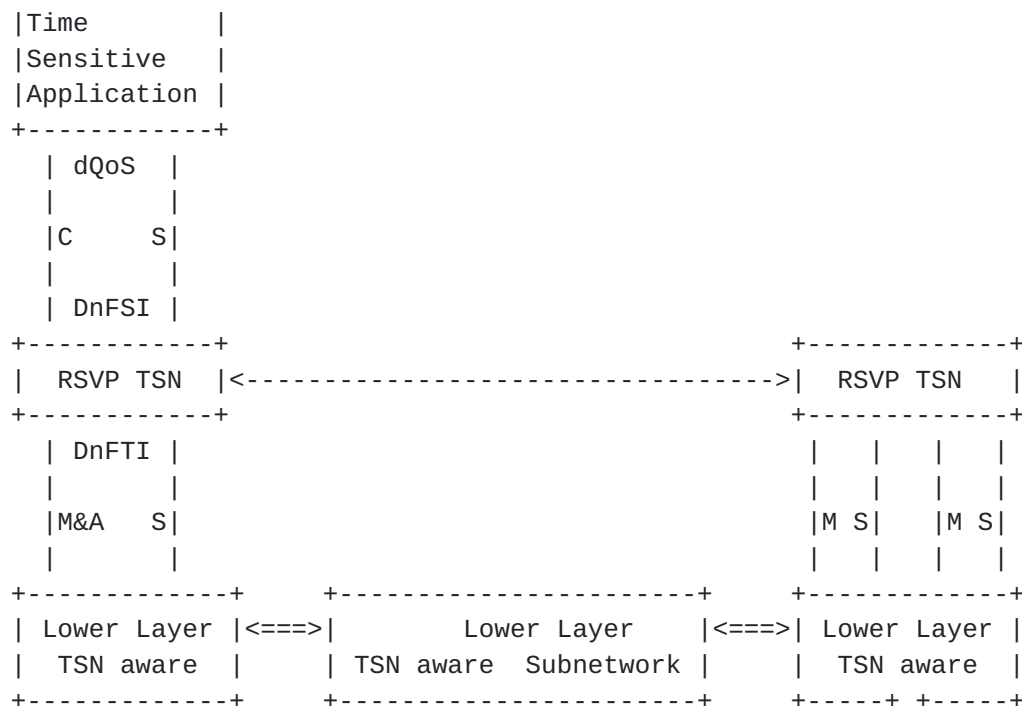
The lower API of RSVP TSN is called DnFlow-Transport-Interface (DnFTI) in this contribution. The DnFTI has connectivity with the lower network layer, which in turn provides deterministic services within a subnetwork based on the TSN model.

For instance, IEEE 802.1Q with extensions for TSN establish streams to transport DnFlows. For stream reservation the Resource-Allocation-Protocol (RAP) [RAP IEEE] has defined the Reservation-Service-Interface (RSI).

The following figure illustrates the information flow within a DetNet end system and a DetNet relay node for the establishment of deterministic end-2-end services.







<---> DnFlow Signaling Service  
 <===> Lower layer transport stream/flow reservation service  
 <===> TSN Stream Reservation  
 dQoS Deterministic QoS time sensitive application interface  
 DnFSI DnFlow-Service-Interface (upper API of RSVP TSN)  
 DnFTI DnFlow-Transport-Interface(lower API of RSVP TSN)  
 C Control  
 S Signaling  
 M&A Maps and Aggregation

Figure 4: Layer Interactions between RSVP TSN and lower layer network supporting TSN

#### 4.2. API for Deterministic QoS (dQoS)

The description of a generalized API to support deterministic QoS is not part of this document.

#### 4.3. DnFlow Signaling Interface (DnFSI)

The definition of the DnFSI and the DnFTI is based on the DnFlow information model [[ID-detnet-flow-information-model](#)].

This interface is oriented on the interface specified by RSVP-IntServ ([RFC 2205](#)). Most of the changes are due to mapping resource reservation styles (see [Section 3.2](#)).



Sender

Call: Open Session (oriented to the RSVP-IntServ interface)

Request parameter (make use of pieces from the DnFlowSpecification)

- DestinationIpAddress, Protocol, DestinationPort

Response parameter:

- SessionID

Call: Add DnFlow

Request parameter (make use of pieces from the DnFlowSpecification)

- SessionID, SourceIpAddress, SourcePort, DSCP
- DnTrafficSpecification: Interval, MaxPacketsPerInterval, MaxPayloadSize, MinPayloadSize
- DnFlowRank
- Select one of the Resource Style: Distinct, Shared, CoordinatedShared
- Data TTL, PATH MTU size, LossRate

Notes for new parameter:

The DSCP is required to map DnFlows according their service class to offered service classes of the lower layer.

The resource style for an DnFlow is announced by the sender within the path message.

The LossRate is accumulated per DnFlow from Sender to Receiver.

Upcall: DnFlow

- Session ID
- One of the Info\_type: RESV\_EVENT; PATH\_ERROR

Receiver



## Call: Open Session

Request parameter (make use of pieces from the DnFlowSpecification)

- DestinationIpAddress, Protocol, DestinationPort

Response parameter

- SessionID

## Call: Join DnFlow

Request parameter

- SessionID
- Select one of the DnFlow Source Selection: Wildcard, List of explicit sources with SourcePort
- MaximumPacketSize
- Extended Traffic Specification: MaximumExpectedLatency

Notes for new parameter:

The Source Selection is split from the RSVP-IntServ Reservation Style but still follows the rules defined by RSVP-IntServ.

The extended traffic specification MaximumExpectedLatency is propagated and merged to a minimum upstream from receiver to sender.

## Upcall: DnFlow

- SessionID
- SourceIpAddress (Sender)
- SourcePort
- One of the Info\_type: RESV\_EVENT; PATH\_ERROR

## General

## Call: Close Session

Request parameter



- SessionID

#### **4.4. DnFlow Transport Interface (DnFTI)**

Sender

Call: Add DnFlow

Request parameter

- SessionID, Interface, DnFlowID, DestinationIpAddress, DSCP
- DnTrafficSpecification: Interval, MaxPacketsPerInterval, MaxPayloadSize, MinPayloadSize, MinPacketsInterval
- One of the Resource Styles: Distinct, Shared, Coordinated Shared

Response parameter

- TransportFlowID (TSN StreamID)

Notes for new parameter:

The DnFlowID is a local parameter to correlate DnFlows to transport flows (e.g., TSN Stream).

The TransportFlowID correlates the DnFlow to the lower layer transport flow, e.g., TSN Stream ID.

Upcall: DnFlow

Response parameter

- SessionID
- TransportFlowID
- One of the Info\_type: RESV\_EVENT, RES\_MODIFY\_EVENT

Receiver

Call: Join DnFlow

Request parameter

- SessionID, Interface, DnFlowID, TransportFlowID



- MaximumPacketSize
- Extended Traffic Specification: MaximumExpectedLatency

Notes for new parameter:

(see notes above)

Upcall: DnFlow

Response parameter

- SessionID, TransportFlowID
- One of the Info\_type: ANNOUNCE\_EVENT, ANNOUNCE\_MODIFY\_EVENT

#### **4.5. RSVP TSN Message Formats**

TBD

### **5. Security Considerations**

Editor's note: This section needs more details.

### **6. IANA Considerations**

N/A

### **7. Conclusion**

This draft outlines recommended changes to RSVP signaling in the form of RSVP TSN for a better alignment of the Layer 3 signaling with that of emerging Layer 2 solutions, together with suggested API specifications for the realization of the L3 to L2 interfaces in endpoints.

### **8. References**

#### **8.1. Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", [RFC 8655](#), DOI 10.17487/RFC8655, October 2019, <<https://www.rfc->



editor.org/info/rfc8655>.

- [RFC2212] Shenker, S., Partridge, C., and Guerin, R., "Specification of Guaranteed Quality of Service", [RFC 2212](#), September 1997.
- [RFC2205] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jasmin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", [RFC 2205](#), September 1997.
- [RFC8655] N. Finn, B. Thubert, B. Vargas, J. Farkas, "Deterministic Networking Architecture", [RFC8655](#), October 2019
- [RFC8938] B. Varga, Ed, J. Farkas, L. Berger, A. Malis, S. Bryant, "Deterministic Networking (DetNet) Data Plane Framework", [RFC8938](#), November 2020.

## 8.2. Informative References

- [ID.malis-detnet-controller-plane-framework] A. Malis, X. Geng, M. Chen, F. Qin, B. Varga, "Deterministic Networking (DetNet) Controller Plane Framework", [draft-malis-detnet-controller-plane-framework-05](#) (work in progress), 2020
- [ID-detnet-flow-information-model] Balazs Varga, Janos Farkas, Rodney Cummings, Yuanlong Jiang, Don Fedyk, "DetNet Flow and Service Information Model", [draft-ietf-detnet-flow-information-model-14](#) (work in progress), 2021
- [CHEN-IEEE] F. Chen, F.J. Goetz, M. Kiessling, J. Schmitt, "Support for uStream Aggregation in RAP (ver 0.3)" (work in progress), Jan 2019, <<http://www.ieee802.org/1/files/public/docs2019/dd-chen-flow-aggregation-0119-v03.pdf>>
- [RAP\_IEEE] IEEE, "P802.1Qdd - Resource Allocation Protocol", (work in progress), <<https://1.ieee802.org/tsn/802-1qdd/>>
- [ID-trossen-detnet-control-signaling] D. Trossen, F.-J. Goetz, J. Schmitt, "DetNet Control Plane Signaling", [draft-trossen-detnet-control-signaling-01](#) (work in progress), 2021



Authors' Addresses

Dirk Trossen  
Huawei Technologies Duesseldorf GmbH  
Riesstr. 25C  
80992 Munich  
Germany

Email: Dirk.Trossen@Huawei.com

Franz-Josef Goetz  
Siemens AG  
Gleiwitzer-Str. 555  
90475 Nuremberg  
Germany

Email: franz-josef.goetz@siemens.com

Juergen Schmitt  
Siemens AG  
Gleiwitzer Str. 555  
90475 Nuremberg  
Germany

Email: juergen.jues.schmitt@siemens.com

