

Deterministic Networking Working Group
Internet-Draft
Intended status: Informational
Expires: 15 September 2026

P. Liu
China Mobile
Y. Li
Huawei
T. Eckert
Futurewei Technologies USA
Q. Xiong
ZTE Corporation
J. Ryoo
ETRI
S. Zhu
New H3C Technologies
X. Geng
Huawei
14 March 2026

Requirements for Scaling Deterministic Networks
draft-ietf-detnet-scaling-requirements-10

Abstract

To support the scaling of deterministic networks, this document describes the technical and operational requirements for networks that exhibit large hop-to-hop latency variation, a large number of flows, and/or multiple domains that do not share a common time source. Applications with varying levels of determinism coexist and are transported in such a network. This document also describes the corresponding Deterministic Networking (DetNet) data plane enhancement requirements.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on 15 September 2026.

Copyright Notice

Copyright (c) 2026 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
2. Conventions Used in This Document	4
3. Technical Requirements in Scaling Deterministic Networks . .	5
3.1. Tolerate Time Asynchrony	5
3.1.1. Support Asynchronous Clocks Across Domains	5
3.1.2. Tolerate Clock Jitter and Wander within a Synchronous Clock Domain	6
3.1.3. Provide Mechanisms not Requiring Strict Time Synchronization	6
3.1.4. Provide Mechanisms not Requiring Synchronization . .	6
3.2. Support Large Single-hop Propagation Latency	7
3.3. Accommodate Higher Link Speeds	8
3.4. Be Scalable to a Large Number of Flows and Tolerate High Utilization of Bandwidth	9
3.5. Tolerate Failures of Links or Nodes and Topology Changes	10
3.6. Prevent Flow Fluctuations	11
3.7. Be Scalable to a Large Number of Hops with Complex Topology	12
3.8. Support Multiple Mechanisms in Single and Multiple Domains	12
3.8.1. Support Provisioning of Multiple Mechanisms	14
3.8.2. Support Switchover Between Mechanisms Across Multiple Domains	15
4. Data Plane Enhancement Requirements	15
4.1. Support Aggregated Flow Identification	16
4.2. Support Information Required by Functions Ensuring Deterministic Latency	16
5. Conclusion	17
6. Security Considerations	17
7. IANA Considerations	17
8. Acknowledgements	17

9. Contributors	17
10. References	18
10.1. Normative References	18
10.2. Informative References	18
Appendix A. Examples of Scaling Deterministic Network Trials . .	20
Authors' Addresses	21

1. Introduction

Packet networks are evolving from bandwidth-guaranteed Quality of Service (QoS) to latency-guaranteed QoS, which ensures both bounded and definite latency. Bounded latency and definite latency can be interpreted as in-time delivery (a packet arrives within a predetermined time) and on-time delivery (a packet arrives exactly at a predetermined time). In addition, network survivability, which has traditionally guaranteed traffic recovery within 50 ms in the event of a network failure, is evolving toward lossless recovery. To support this evolution of QoS and network survivability, Time-Sensitive Networking (TSN) and Deterministic Networking (DetNet) technologies are considered essential.

TSN, a set of standards developed by the IEEE 802.1 TSN Task Group (TG) [IEEE802.1TSN], specifies mechanisms and protocols necessary to realize highly available IEEE 802.1 networks with bounded latency to carry time-sensitive, real-time application traffic.

DetNet, whose architecture is defined in RFC 8655 [RFC8655], supports the transport of specified unicast or multicast data flows for real-time applications with extremely low data loss rates and bounded latency, operating under a single administrative control or within a closed group of administrative control. The overall framework for the DetNet data plane is described in [RFC8938], and several documents have been standardized for various data plane technologies and their interworking, extending TSN's service capabilities to IP (Internet Protocol) and MPLS (Multi-Protocol Label Switching) networks.

As deterministic networks scale or span multiple interconnected domains, DetNet solutions must consider one or more of the following aspects:

- * Clock synchronization across different domains may be relaxed or entirely absent. (Section 3.1)
- * The end-to-end path may comprise both low- and high-latency hops. (Section 3.2)

- * Various transmission rates may be supported across different ports and network nodes. (Section 3.3)
- * A large number of flows may make per-flow state identification difficult and lead to high bandwidth utilization. (Section 3.4)
- * Topology changes and link or node failures may occur more frequently. (Section 3.5)
- * Flow fluctuations caused by the large number of flows may occur frequently. (Section 3.6)
- * The end-to-end path may include a large number of hops and involve complex topologies. (Section 3.7)
- * Mechanisms used to ensure bounded latency (e.g., queuing methods) may vary or be differently configured across multiple domains. (Section 3.8)

Such domains are typically under a single administrative control or consist of multiple cooperating administrative networks within a closed group [RFC8655]. However, they may belong to different Autonomous System (AS) domains and be controlled by multiple peering or cascaded controllers, and they may not share the same time source. Maintaining per-flow status becomes impractical in a large-scale network. Aggregation and disaggregation of flows occur both at the boundaries and within DetNet domains, governed by various rules. Flow identification and packet treatment should address individual or combined changes introduced by the scaling of deterministic networks.

Based on the considerations above, this document describes the requirements for scaling deterministic networks. Scaling deterministic networks demands enhancements to the existing bounded latency mechanisms and the corresponding data plane in order to ensure DetNet service across a single administrative network or multiple cooperating administrative networks, as defined in the DetNet charter. Deterministic networking for the open Internet is outside the current scope.

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119][RFC8174] when, and only when, they appear in all capitals, as shown here.

While [RFC2119] and [RFC8174] describe interpretations of these key words in terms of protocol specifications and implementations, they are used in this document to describe technical and operational requirements to support the scaling of deterministic networks.

3. Technical Requirements in Scaling Deterministic Networks

Given the attributes described in Section 1, the corresponding technical requirements should be taken into account.

3.1. Tolerate Time Asynchrony

A deterministic network may span multiple networks with one or more cooperating administrative domains. The presence of diverse network nodes in these domains can result in disparate local time variations, which require tolerance for time asynchrony.

3.1.1. Support Asynchronous Clocks Across Domains

One of DetNet's objectives is to stitch TSN islands together. All devices within a TSN domain are time-synchronized, and most TSN technologies rely on precise time synchronization [IEEE802.1Qbv][IEEE802.1Qch][IEEE802.1Qav]. However, different TSN islands may operate with unsynchronized clocks, as shown in Figure 1, where the time difference between two TSN domains is denoted as D . DetNet needs to connect these two TSN domains and provide end-to-end deterministic latency services. The mechanism adopted by a deterministic network MUST be prepared to traverse multiple unsynchronized TSN domains. This can be achieved, for example, by adding extra buffer space at the ingress of a new domain, increasing the dead time as a guard band [IEEE802.1Qdv], or using some timing compensation mechanisms. This document does not intend to provide an exhaustive list of such methods.

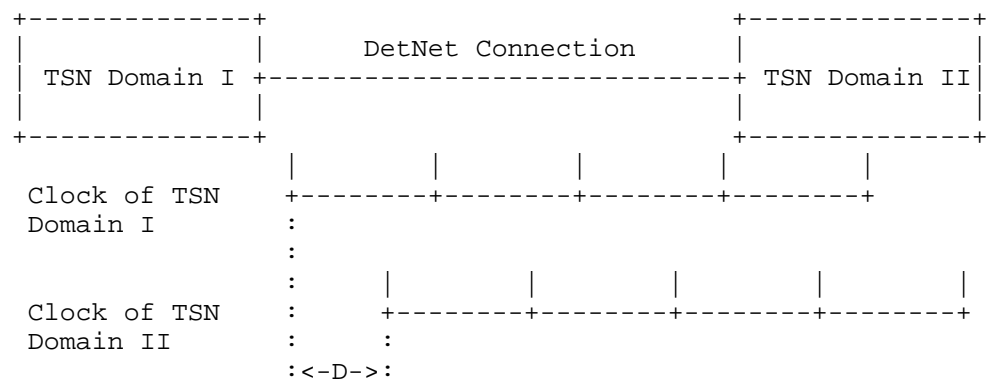


Figure 1: Clock asynchrony between two TSN islands

3.1.2. Tolerate Clock Jitter and Wander within a Synchronous Clock Domain

Within a single time synchronization domain, varying levels of clock accuracy are expected. For example, the crystal oscillator used in Ethernet is specified at 100 ppm [Fast-Ethernet-MII-clock], Synchronous Ethernet (SyncE) can achieve 50 ppb [G.8262], and even higher timing accuracy is expected in 5G mobile backhaul [G.8273]. These clocks experience different levels of jitter and wander, which may result in different levels of path asymmetry. Deterministic networks SHOULD be capable of compensating for or absorbing such time variations within a domain.

3.1.3. Provide Mechanisms not Requiring Strict Time Synchronization

It is usually difficult to achieve full time synchronization as the scale of the networks increases. Some networks, such as mobile backhaul, use frequency synchronization (e.g., SyncE) instead of strict time synchronization. The same deterministic performance in terms of bounded latency and jitter SHOULD be achieved even when full time synchronization is not available, i.e., when only partial synchronization is in use, with SyncE being one example.

3.1.4. Provide Mechanisms not Requiring Synchronization

A deterministic network can contain a large number of traffic flows, some of which are non-periodic. Asynchronous methods can meet the requirements of such traffic flows. Moreover, mechanisms that do not require time and/or frequency synchronization can reduce hardware cost and complexity at network nodes. [IEEE802.1Qcr] conceptually introduces per-flow-based asynchronous shaper to achieve bounded

latency. The effectiveness of the per-flow-based asynchronous shaper can be demonstrated through mathematical analysis. It naturally tolerates time variation, but raises concerns about per-flow state buffer management. When it is in use, the requirement in Section 3.3 SHOULD be carefully considered.

3.2. Support Large Single-hop Propagation Latency

In some deterministic networks, a single hop can introduce significant latency. Optical transmission in fiber travels at approximately 200 km/ms. Thus, the propagation delay of a single hop can be on the order of a few milliseconds. This is much greater than the single hop propagation delays found in typical Local Area Networks (LANs), and can have a substantial impact on queuing mechanisms, such as cyclic or time-aware scheduling methods. Therefore, propagation delay must be taken into account in end-to-end computations. For example, it should be considered when setting the period in both time- and frequency-synchronized methods, or when setting the extra buffer in the asynchronous methods, to meet the requirements of deterministic forwarding between the network nodes.

Here, we use an example to illustrate the impact of large single-hop propagation delay on cycle-based methods, without proposing any specific solution. In a cycle-based method, suppose a deterministic network aims to maintain a simple cycle mapping relationship, but the link distance between two nodes is relatively long. Moreover, a downstream node may have multiple upstream nodes with different link propagation delays (e.g., 9 us, 10 us, 11 us, 15 us and 20 us). In order to absorb the longest link propagation delay, the length of the cycle must be set to more than 20 us. In [IEEE802.1Qch], propagation delay is part of the dead time imposed within a cycle, which negatively affects bandwidth utilization. However, since packet arrival times can vary within the receiving cycle, a longer cycle leads to greater delay variation.

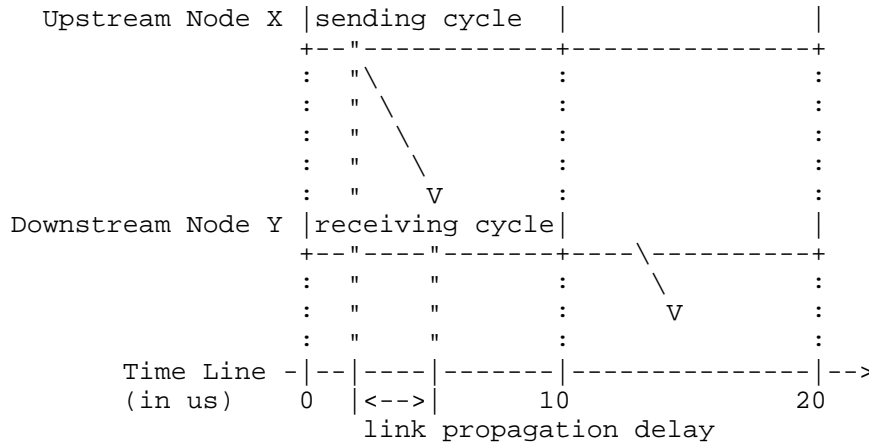


Figure 2: The influence of link propagation delay on a cyclic method

Note that Figure 2 is provided solely as an example to illustrate latency caused by large single-hop propagation. CQF is not limited to two queues. Instead, using more than two queues can be an option to address latency-related concerns associated with long links. [Multiple-CQF] provides a more detailed discussion of this problem and proposes a mechanism to address it.

3.3. Accommodate Higher Link Speeds

A deterministic network can use higher-speed links, especially for its backbone. At the time of publication, deterministic mechanisms in local networks typically operate at link speeds of 10 Mbps, 1 Gbps, or possibly 10 Gbps. In contrast, data rates of 10Gbps, 100Gbps, 400Gbps, and even higher are commonly used in wide area networks. With increasing data rates, the network scheduling cycle can be shortened if the same amount of data is required to be sent in each cycle for each application. Alternatively, more data can be sent if the cycle time remains the same. The former case requires more precise time control, such as cycles on the order of a few microseconds or even sub-microseconds, for the input stream gate and the timed output buffer. The latter requires more buffer space, which imposes more complex buffer and queue management and greater memory consumption.

Another aspect to consider is the aggregation of flows. In a deterministic network, the number of flows can reach hundreds or even tens of thousands. These flows can be aggregated into a small number of deterministic paths or tunnels. It is practical to maintain a small amount of flow-based or aggregated-flow-based state in local

networks. However, in higher-speed and larger-scale networks, this becomes much less feasible. If [IEEE802.1Qcr] is in use, it requires more buffers compared to other mechanisms that are fully or partially time-synchronized. Therefore, further optimization is needed to support higher link speeds.

3.4. Be Scalable to a Large Number of Flows and Tolerate High Utilization of Bandwidth

A deterministic network may carry a large number of traffic flows. According to [RFC2475], per-flow state identification in the network becomes infeasible as flow count scales up. As a result, identifying individual IP flows in the DetNet data plane or configuring per-flow state at each node becomes nearly impossible at scale. DetNet supports flow aggregation. However, in large-scale networks, individual flows may dynamically join or leave aggregated flows, which causes instability in the identification of these aggregated DetNet flows. Wildcards and value ranges used for flow identification may need to be adjusted to ensure that aggregated flows maintain consistent deterministic characteristics.

For scaling networks, proper provisioning at the control plane is required to accommodate higher levels of aggregation. Provisioning on aggregated flows normally improves scalability, but at the cost of coarse-grained filtering and protection of incoming traffic. It is desirable that adding a flow to the network does not affect the latency of existing flows or require complex recalculations, and instead requires as little work as possible. For instance, filtering and policing configurations are applied only at ingress nodes, not at intermediate ones. [IEEE802.1Qbv] uses traffic classes to divide flows, typically limited to eight. As a result, the forwarding mechanism itself remains relatively simple even with a large number of flows or a higher degree of aggregation. However, when new flows are added, the Gate Control List may need to be updated, and the associated recalculation can be complex. There may be methods to simplify the calculation or configuration, but additional work is needed to enhance them.

Meanwhile, traffic requiring deterministic networking can significantly utilize the capacity of a link, or the portion of the link that is dedicated to such traffic, for example, exceeding 75% or even approaching full (near 100%) utilization. Typically, resources required for DetNet are reserved. However, over-provisioning of link capacity may not work in such cases. To guarantee deterministic latency and jitter in such environments, it is preferable to provide scalable queuing solutions that improve bandwidth utilization, based on existing methods, including TSN and other published standards. For instance, when the bandwidth utilization is high, the guard band in each cycle in [IEEE802.1Qch] is a type of over-provisioning and can be optimized through more scalable queuing enhancements.

3.5. Tolerate Failures of Links or Nodes and Topology Changes

Deterministic networks may involve a greater number of network devices, and the addition or removal of such devices tends to occur more frequently than in LANs. A representative use case is ultra-low-latency public 5G transport networks, which would require DetNet to extend to every 5G base station. For some network operators, their networks may need to connect approximately 100,000 base stations, serving multiple mobile network operators. This number is expected to increase further as 5G deployment continues.

On the one hand, a large number of network devices may increase the likelihood of network link failures. Path switching or routing reconvergence can result in significant packet loss and retransmission delays, often lasting several seconds before the network stabilizes. As the delays involved are too large to be mitigated by jitter compensation techniques, it is necessary to support mechanisms that can adapt to link or node failures and topology changes.

On the other hand, changes in the number of devices may affect the implementation and adjustment of deterministic networking mechanisms, such as topology discovery, queuing mechanisms, and packet replication and elimination. For instance, having multiple fully disjoint paths when implementing the Packet Replication, Elimination, and Ordering Functions (PREOF) increases survivability in the event of a node or link failure on any path. However, this also introduces challenges in finding paths with similar lengths and/or hop counts, ensuring sufficient buffer space to absorb latency differences between paths at scale. Therefore, a method is needed to support flexible multi-path planning and provide a reliable foundation for implementing PREOF.

3.6. Prevent Flow Fluctuations

A wider variety of DetNet traffic flows described in Section 3.4 will lead to more frequent dynamic joining and leaving of flows. This, in turn, increases flow fluctuations and the overall unpredictability of DetNet traffic. Examples include the following:

- * Various high-volume traffic flows of different applications in scaling networks easily cause more bursty traffic.
- * An increasing number of aggregation nodes receive flows from more upstream nodes, introducing additional nondeterministic delay in packet handling.
- * Bursts of flows accumulate as the flows traverse, merge, and diverge across multiple hops. Even a minor error in packet treatment at one node can have a cascading effect on downstream nodes.
- * Loops formed in the network topology increase the maximum bursts of flows exponentially [ANDREWS][BOUILLARD][THOMAS].
- * Node and link failures, which are more common in large-scale networks (Section 3.5), require dynamic traffic steering to alternate paths. This can further contribute to flow fluctuations.

It should be noted that non-DetNet flows can also be high in volume and may impact the scalability of DetNet flows. For example, they may lead to high bandwidth utilization, limiting the ability to reserve resources or to perform effective traffic steering for DetNet flows. However, it is assumed that appropriate strategies will be deployed at the ingress to manage non-DetNet traffic and prevent real-time interference with DetNet traffic.

Support for bursty traffic is essential, along with mechanisms to mitigate micro-bursts. To accommodate flow fluctuations, pre-planned configurations, ingress traffic conditioning, scalable queuing, and enhanced buffer capacity are necessary. In addition, the time required for network reconfiguration to respond to such changes should be strictly controlled. For example, it may need to be limited to a specified threshold.

3.7. Be Scalable to a Large Number of Hops with Complex Topology

Scaling networks often results in situations where an end-to-end flow traverses a large number of hops, for example, 15 or more. The network topology can also be complex, including star, ring, mesh, and their combinations, and may be hierarchical. It is required to support networks with such diverse topologies and large hop counts.

Delivering DetNet QoS in large and complex networks requires end-to-end bounds on both latency and jitter, as discussed in Section 3.1 of [RFC8655]. Achievable end-to-end latency bounds necessarily depend on the number of hops for a flow. In the best case, the additional latency introduced by queuing mechanisms for DetNet QoS is bounded by a fixed per-hop maximum value, making the resulting end-to-end latency bounds a linear function of the number of hops in addition to the inherent forwarding latency of the nodes involved. In contrast, it is possible to achieve fixed end-to-end jitter bounds that are independent of the number of hops. Such fixed jitter bounds are strongly preferable to those that increase with hop count.

DetNet QoS requires resource allocation in advance (e.g., link bandwidth and node buffer resources), as discussed in Section 3.2.1 of [RFC8655]. The complexity of resource allocation can range from linear (e.g., allocating resources for each hop via a path-based resource reservation protocol such as RSVP [RFC2205]) to potentially exponential (e.g., if solving a complex graph optimization problem is required). Although this complexity does not directly affect the achievable end-to-end latency and jitter bounds, it does influence other aspects such as the computational effort and the time required to admit a new flow without disrupting the QoS of existing DetNet flows.

Different queuing mechanisms exhibit different properties in terms of achievable end-to-end jitter bounds, achievable end-to-end latency bounds, and the complexity of resource allocation. All three factors should be considered in the evaluation and selection of scalable DetNet queuing mechanisms.

3.8. Support Multiple Mechanisms in Single and Multiple Domains

As described in Section 3.4, there will be a large number of flows, each potentially having a different level of demand for DetNet services. [RFC8578] provides various use cases and their requirements in the areas of industry, electricity, buildings, etc. Some of these use cases clearly specify requirements for both latency and jitter, while others omit jitter requirements. Different types of users have different demands, just as a network provider offers different network services for personal or enterprise business.

Some applications have critical Service Level Agreement (SLA) requirements, such as remote control in manufacturing, cloud-based Programmable Logic Controllers (PLCs) for industrial automation, and manufacturing and differential protection in power systems. For these services, exceeding latency or jitter bounds can result in property loss or security risks. Therefore, users of these services cannot tolerate any non-deterministic behavior and are typically willing to pay more for higher-quality network services. Other applications have more relaxed SLA requirements, such as cloud gaming, cloud-based virtual reality (VR), and online meetings in "consumer" networks. While users of these services seek a good quality of experience, they can tolerate some level of performance variation. For example, occasional violations of latency bounds may be acceptable if they occur infrequently. Those different applications expect different types of solutions, often corresponding to varying cost levels.

Different SLA requirements and the presence of multiple domains in scaling networks necessitate the use of diverse DetNet technologies and queuing mechanisms. For services that demand strict determinism, highly deterministic queuing technologies need to be deployed, which may require upgrading all network devices. In contrast, for less stringent services, it may be sufficient to upgrade only certain devices, such as edge nodes, or to share existing network resources. As more queuing mechanisms are developed, it becomes increasingly desirable to provide queuing solutions that support multiple levels of deterministic performance and to allocate resources appropriately to optimize overall network resource utilization. These different queuing technologies may be used in the following scenarios:

- * Within the same network, where some devices are equipped with multiple queuing technologies. This allows the operator to select the appropriate service type or level as needed.
- * Across multiple network domains, where different domains support different queuing technologies and coordination between them is required.
- * In separate network implementations tailored for distinct application domains, where no additional coordination is necessary.

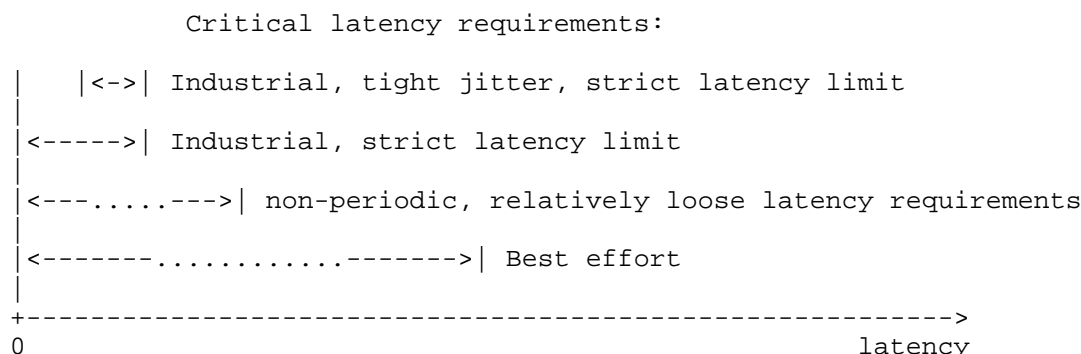


Figure 3: Different levels of application requirements

3.8.1. Support Provisioning of Multiple Mechanisms

A deterministic network should support a variety of deterministic services to meet the diverse requirements of various applications. This includes supporting corresponding queuing mechanisms at multiple DetNet QoS levels, if necessary. For example, different queuing mechanisms can offer varying levels of latency, jitter, and other guarantees, and a single network device may implement multiple mechanisms concurrently. An aggregation device, for instance, may employ mechanisms defined in [IEEE802.1Qbv] and [IEEE802.1Qcr], and other mechanisms to forward traffic along different paths simultaneously. The need to support multiple queuing mechanisms becomes especially prominent in large-scale networks, compared to small-scale environments. In such cases, more than eight queues or sub-queues may be required to support complex queuing strategies, exceeding the typical eight traffic classes available in TSN-enabled networks.

Accordingly, configuring multiple queuing mechanisms in deterministic networks can be complex. Such configurations MUST support unified or simplified approaches to scheduling and managing multiple queuing mechanisms. For example, in distributed scenarios without a controller, information about the queuing mechanisms, including types and associated algorithms, and queue forwarding capabilities with different levels of latency and jitter guarantees, can be advertised within the domain. In centralized scenarios, this information can be reported to the controller, allowing it to build a deterministic network resource topology pool for path computation. In addition, to enable fine-grained resource management and ensure resource guarantees during session setup and teardown, it is necessary to establish standardized methods for quantifying and measuring resources associated with interfaces and queues.

3.8.2. Support Switchover Between Mechanisms Across Multiple Domains

In deterministic networks, the end-to-end service may span multiple network domains. Each domain may adopt different queuing mechanisms or may operate at different link speeds (see Section 3.3) for the same queuing mechanism.

Both cases may generate additional end-to-end latency, jitter, and packet loss, as different queuing mechanisms and link speeds may result in varying scheduling granularities or phases between domains. In the case of a queuing mechanism switchover, such as from a time-synchronized mechanism like [IEEE802.1Qbv] to an asynchronous mechanism like [IEEE802.1Qcr], a collaboration mechanism across multiple domains MUST be considered. Similarly, when switching between different link speeds, such as from 1 Gbps to 10 Gbps (or vice versa), the quantification of deterministic forwarding resources (such as time slots) of the queuing mechanism MUST be aligned with the link speed.

A device that connects multiple domains requires a flexible queue stitching function. This may include increasing buffer capacity at inter-domain devices to provide sufficient adjustment space when flows are forwarded across different queuing mechanisms, applying jitter compression to decouple queuing behaviors between domains, or using additional traffic shaping solutions to smooth traffic. These techniques help ensure that the same scale of service flows can be forwarded successfully across domains that use different queuing mechanisms and/or operate at different link speeds.

4. Data Plane Enhancement Requirements

According to [RFC8938], the DetNet data plane can provide or carry two metadata items in MPLS and IP data planes: Flow-ID and sequence number. The Flow-ID is used to identify individual DetNet flows or aggregated flows, while the sequence number is used by PREOF for each DetNet flow. The Flow-ID is used by both the service and forwarding sub-layers, whereas the sequence number is only used by the service layer. Metadata can also be used for OAM indications and instrumentation related to DetNet data plane operation.

In general, support for additional data plane metadata and related processing SHOULD be provided in deterministic networks. With the introduction of IPv6 Extension Headers [RFC8200] and Segment Routing over IPv6 [RFC8986], native IPv6 data plane SHOULD be supported along with the IPv6-specific enhancements. This section lists the data plane enhancement requirements that are based on, but not limited to, the technical requirements in Section 3. It describes how IP and/or MPLS, along with related OAM, can be used to support flow

identification and packet treatment over Layer 3. Some enhancements to the control plane may also be required to meet the requirements in Section 3. However, these are out of scope for this document and are expected to be addressed in [I-D.ietf-detnet-controller-plane-framework] or other documents.

4.1. Support Aggregated Flow Identification

Current IPv6 aggregated flow identification is generally based on 5 or 6 tuples, IP prefixes, or wildcards as indicated in [RFC8938]. However, in deterministic networks, the number of individual flows can be huge, and these flows may dynamically join or leave an aggregated flow at each hop, as described in Section 3. Such behaviors lead to the difficulty in identifying aggregated flows by relying on prefixes or wildcards.

In addition, deterministic services may demand different deterministic QoS requirements according to different levels of application requirements. To address this, flow identification with service-level aggregation should be supported. Flow identification also enables packets to be quickly directed to the appropriate queue. In scaling networks, a large number of flows may require either deterministic latency or normal forwarding services. Explicit flow identification makes it easier to quickly distinguish the DetNet flows without relying on the longest-match rule across multiple header fields in IP data plane. Therefore, explicit aggregated flow identification SHOULD be supported.

4.2. Support Information Required by Functions Ensuring Deterministic Latency

According to Section 3.1, deterministic networks should support synchronized or asynchronous queuing mechanisms. Different queuing mechanisms require different types of DetNet-specific metadata to support functions, such as traffic regulation and queue management, which are used to ensure deterministic latency. For instance, the data plane needs to support the identification of the cycle for cyclic queuing and forwarding, or latency-related information for time-based queuing, in order to enable the use of corresponding queuing and/or scheduling mechanisms in a large-scale network.

When different queuing mechanisms are deployed at the same network node, the corresponding metadata for each queuing mechanism should be provided simultaneously. When multiple types of metadata are carried in a single packet, the metadata format should be self-descriptive and extensible to support a variable number of metadata fields. However, carrying additional metadata increases processing overhead, such as requiring fixed- or variable-length lookups, and increasing

the number of read/write operations on packet headers. Therefore, data plane processing efficiency also needs to be considered when ensuring deterministic latency. The specific methods or solutions are out of scope of this document.

This document does not specify what metadata and formats should be carried in the data plane. Solution documents should clearly explain why and how the metadata carried as the data plane interacts with queuing or other functions, and how it contributes to deterministic network deployment.

5. Conclusion

This document specifies the technical requirements for scaling deterministic networks and enhancing the corresponding data plane. Some of the proposed queuing mechanisms and trials are cited, as they provide useful insights for improving existing queuing mechanisms to meet the requirements of scaling deterministic networks.

6. Security Considerations

When DetNet flows span multiple domains and require multi-domain collaboration, security guarantees need to be strengthened.

7. IANA Considerations

No IANA actions are requested.

8. Acknowledgements

The authors would like to thank David Black, Jino Joun, Lou Berger, Balazs Varga, Fan Yang, Tianran Zhou, and Yaakov Stein for their helpful suggestions. The authors also would like to thank Liang Geng, Peter Willis, Shunsuke Homma, and Li Qiang for their previous works.

9. Contributors

The following people have contributed substantially to this document:

Zongpeng Du
China Mobile
Email: duzongpeng@chinamobile.com

Lei Zhou
New H3C Technologies
Email: zhou.leih@h3c.com

10. References

10.1. 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>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [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>>.
- [RFC8938] Varga, B., Ed., Farkas, J., Berger, L., Malis, A., and S. Bryant, "Deterministic Networking (DetNet) Data Plane Framework", RFC 8938, DOI 10.17487/RFC8938, November 2020, <<https://www.rfc-editor.org/info/rfc8938>>.
- [RFC8964] Varga, B., Ed., Farkas, J., Berger, L., Malis, A., Bryant, S., and J. Korhonen, "Deterministic Networking (DetNet) Data Plane: MPLS", RFC 8964, DOI 10.17487/RFC8964, January 2021, <<https://www.rfc-editor.org/info/rfc8964>>.

10.2. Informative References

- [ANDREWS] M, A., "Instability of FIFO in the permanent sessions model at arbitrarily small network loads. ACM Trans. Algorithms, vol. 5, no. 3, pp. 1-29, doi:10.1145/1541885.1541894.", July 2009.
- [BOUILLARD] Corronc, B. A. B. M. A. E. L., "Deterministic network calculus: From theory to practical implementation. in Networks and Telecommunications. Hoboken, NJ, USA: Wiley, doi: 10.1002/9781119440284.", January 2018.
- [Fast-Ethernet-MII-clock] "Fast Ethernet MII clock".
- [G.8262] International Telecommunication Union, "Timing characteristics of a synchronous equipment slave clock", ITU-T Recommendation G.8262, November 2018.

[G.8273] International Telecommunication Union, "Framework of phase and time clocks", ITU-T Recommendation G.8273, March 2018.

[I-D.ietf-detnet-controller-plane-framework]

Malis, A. G., Geng, X., Chen, M., Varga, B., and C. J. Bernardos, "A Framework for Deterministic Networking (DetNet) Controller Plane", Work in Progress, Internet-Draft, draft-ietf-detnet-controller-plane-framework-15, 24 September 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-detnet-controller-plane-framework-15>>.

[IEEE802.1Qav]

IEEE, "IEEE Standard for Local and metropolitan area networks -- Virtual Bridged Local Area Networks - Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams", IEEE 802.1Qav-2009, DOI 10.1109/IEEESTD.2010.8684664, 5 January 2010, <<https://doi.org/10.1109/IEEESTD.2010.8684664>>.

[IEEE802.1Qbv]

IEEE, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", IEEE 802.1Qbv-2015, DOI 10.1109/IEEESTD.2016.8613095, 18 March 2016, <<https://doi.org/10.1109/IEEESTD.2016.8613095>>.

[IEEE802.1Qch]

IEEE, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 29: Cyclic Queuing and Forwarding", IEEE 802.1Qch-2017, DOI 10.1109/IEEESTD.2017.7961303, 28 June 2017, <<https://doi.org/10.1109/IEEESTD.2017.7961303>>.

[IEEE802.1Qcr]

IEEE, "IEEE Standard for Local and Metropolitan Area Networks -- Bridges and Bridged Networks - Amendment 34: Asynchronous Traffic Shaping", IEEE 802.1Qcr-2020, DOI 10.1109/IEEESTD.2020.9253013, 6 November 2020, <<https://doi.org/10.1109/IEEESTD.2020.9253013>>.

[IEEE802.1Qdv]

IEEE, "IEEE Standard for Local and metropolitan area networks -- Enhancements to Cyclic Queuing and Forwarding", IEEE 802.1Qdv-2023, 12 July 2023.

[IEEE802.1TSN]

IEEE Standards Association, "IEEE 802.1 Time-Sensitive Networking Task Group",
<<https://www.ieee802.org/1/pages/tsn.html>>.

[Multiple-CQF]

IEEE, "Multiple Cyclic Queuing and Forwarding."
<https://www.ieee802.org/1/files/public/docs2021/new-finn-multiple-CQF-0921-v02.pdf>".

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <<https://www.rfc-editor.org/info/rfc2205>>.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, DOI 10.17487/RFC2475, December 1998, <<https://www.rfc-editor.org/info/rfc2475>>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.

[RFC8578] Grossman, E., Ed., "Deterministic Networking Use Cases", RFC 8578, DOI 10.17487/RFC8578, May 2019, <<https://www.rfc-editor.org/info/rfc8578>>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.

[THOMAS] Mifdaoui, T. L. L. B. J. A. A., "On cyclic dependencies and regulators in time-sensitive networks. IEEE Real-Time Syst. Symp. (RTSS), York, U.K., pp. 299-311.", December 2019.

Appendix A. Examples of Scaling Deterministic Network Trials

Some trials have been conducted to validate the concept of large-scale deterministic networks.

To validate deterministic networking technology in large-scale networks, a trial of Deterministic IP on China Environment for Network Innovations (CENI), which is a network built for new network technology trials, was conducted. The trial covered a distance of 3,000 km over 13 hops, and jitter was controlled within 100 us.

To validate remote control over Deterministic IP with strict performance requirements, a trial was conducted in cooperation with Baosteel, a Chinese steel company that proposed latency requirements of less than 4 ms and jitter under 20 us. The trial network spanned 600 km. Both the first and second trials were based on a frequency synchronization solution.

To realize synchronization of multiple flows across an inter-provincial network during an exhibition, Emergen proposed a requirement in which two flows, one carrying video and the other carrying VR content, would be transmitted from Province A and would arrive at Province B at the same time. This was intended to enable synchronized playback of camera-captured video alongside the corresponding VR model. This requirement was proposed to support the deployment of virtual industry products. Due to time constraints and other limitations, the requirement was fulfilled using edge network devices and was delivered with a lower level of SLA.

ETRI, in collaboration with a smart factory operator, network operators, equipment vendors, and universities, demonstrated an ultra-low-latency, high-reliability 5G wired and wireless network-based remote Industrial Internet of Things (IIoT) service. The demonstration connected a control center and a smart factory across the networks of three different operators over a distance of 280 km. In this trial, it was shown that real-time remote smart manufacturing is feasible, with round-trip delay maintained below 3 ms within the smart factory and below 10 ms between remote 5G industrial devices. In the future, the team plans to examine the feasibility of large-scale deterministic networking by interconnecting smart factories in Gyeongsan, South Korea, and Oulu, Finland.

These trials indicate that both operators and enterprise users increasingly demand deterministic performance in large-scale networks. However, the implementation technologies used to achieve this are not the same across deployments.

Authors' Addresses

Peng Liu
China Mobile
Beijing
100053
China
Email: liupengyjy@chinamobile.com

Yizhou Li
Huawei
Nanjing
210012
China
Email: liyizhou@huawei.com

Toerless Eckert
Futurewei Technologies USA
Santa Clara, 95014
United States of America
Email: tte@cs.fau.de

Quan Xiong
ZTE Corporation
Wuhan
430223
China
Email: xiong.quan@zte.com.cn

Jeong-dong Ryoo
ETRI
Daejeon
34129
South Korea
Email: ryoo@etri.re.kr

Shiyin Zhu
New H3C Technologies
Beijing
100094
China
Email: zhushiyin@h3c.com

Xuesong Geng
Huawei
Beijing
100095
China
Email: gengxuesong@huawei.com