

DetNet  
Internet-Draft  
Intended status: Informational  
Expires: September 22, 2016

J. Korhonen, Ed.  
Broadcom  
J. Farkas  
G. Mirsky  
Ericsson  
P. Thubert  
Cisco  
Y. Zhuang  
Huawei  
L. Berger  
LabN  
March 21, 2016

DetNet Data Plane Protocol and Solution Alternatives  
draft-dt-detnet-dp-alt-00

Abstract

This document identifies existing IP and MPLS, and other encapsulations that run over IP and/or MPLS data plane technologies that can be considered as the base line solution for deterministic networking data plane definition.

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 <http://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 September 22, 2016.

Copyright Notice

Copyright (c) 2016 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

1. Introduction . . . . .	3
2. DetNet Data Plane Overview . . . . .	3
3. Criteria for data plane solution alternatives . . . . .	6
3.1. #? DetNet Service Interface . . . . .	6
3.2. #1 Encapsulation and overhead . . . . .	7
3.3. #2 Flow identification . . . . .	7
3.4. #3 Packet sequencing . . . . .	7
3.5. #4 Explicit routes . . . . .	8
3.6. #5 Packet replication and deletion . . . . .	8
3.7. #6 Operations, Administration and Maintenance . . . . .	9
3.8. #7 Time synchronization . . . . .	9
3.9. #8 Class and quality of service capabilities . . . . .	9
3.10. #9 Packet traceability . . . . .	10
3.11. #10 Technical maturity . . . . .	10
4. Data plane solution alternatives . . . . .	11
4.1. DetNet Transport layer technologies . . . . .	11
4.1.1. Native IPv6 transport . . . . .	11
4.1.2. Native IPv4 transport . . . . .	14
4.1.3. Multiprotocol Label Switching (MPLS) . . . . .	16
4.2. DetNet Service layer technologies . . . . .	21
4.2.1. Generic Routing Encapsulation (GRE) . . . . .	21
4.2.2. Layer-2 Tunneling Protocol (L2TP) . . . . .	24
4.2.3. Virtual Extensible LAN (VXLAN) . . . . .	24
4.2.4. MPLS-based Services . . . . .	24
4.2.5. Pseudo Wire Emulation Edge-to-Edge (PWE3) . . . . .	26
4.2.6. MPLS-Based Ethernet VPN (EVPN) . . . . .	30
4.2.7. Bit Indexed Explicit Replication (BIER) . . . . .	33
4.2.8. Higher layer header fields . . . . .	41
5. Summary of data plane alternatives . . . . .	42
6. Security considerations . . . . .	42
7. IANA Considerations . . . . .	42
8. Acknowledgements . . . . .	42
9. References . . . . .	43
9.1. Informative References . . . . .	43
9.2. URIs . . . . .	52
Appendix A. Examples of combined DetNet Service and Transport layers . . . . .	52
Authors' Addresses . . . . .	52

## 1. Introduction

Deterministic Networking (DetNet) [I-D.finn-detnet-problem-statement] provides a capability to carry unicast or multicast data flows for real-time applications with extremely low data loss rates and known upper bound maximum latency [I-D.finn-detnet-architecture]. The deterministic networking Quality of Service (QoS) is expressed as 1) the maximum end-to-end latency from sender (talker) to receiver (listener), and 2) probability of loss of a packet. Only the worst-case values for the mentioned parameters are concerned.

There are three techniques to achieve the QoS required by deterministic networks:

- o zero congestion loss,
- o explicit routes,
- o packet replication and deletion.

This document identifies existing IP and Multiprotocol Label Switching (MPLS) [RFC3031], layer-2 or layer-3 encapsulations and transport protocols that could be considered as foundations for a deterministic networking data plane. The full scope of the deterministic networking data plane solution is considered including, as appropriate: quality of service (QoS); Operations, Administration and Maintenance (OAM); and time synchronization among other criteria described in Section 3.

This document does not select a deterministic networking data plane protocol. It does, however, elaborate what it would require to adapt and use a specific protocol as the deterministic networking data plane solution. This document is only concerned with data plane considerations and, specifically, with topics that potentially impact potential deterministic networking aware data plane hardware. Control plane considerations are out of scope of this document.

## 2. DetNet Data Plane Overview

[Editor's note: all/portions of the following may be moved to the DetNet Architecture document at some future point.]

A "Deterministic Network" will be composed of DetNet enabled "End Systems", DetNet enabled "Edge Nodes", and DetNet enabled "Network Nodes". DetNet enabled nodes will provide a DetNet service to attached DetNet End Systems. All DetNet enabled systems and nodes will be interconnected by sub-networks. These sub-networks will provide DetNet compatible service for support of DetNet traffic. Examples of sub-networks include 802.1TSN and a point-to-point OTN link. Of course, multi-layer DetNet systems may be possible too,

where one DetNet appears as a sub-network, and provides service to, a higher layer DetNet system. A simple DetNet is shown in Figure 1.

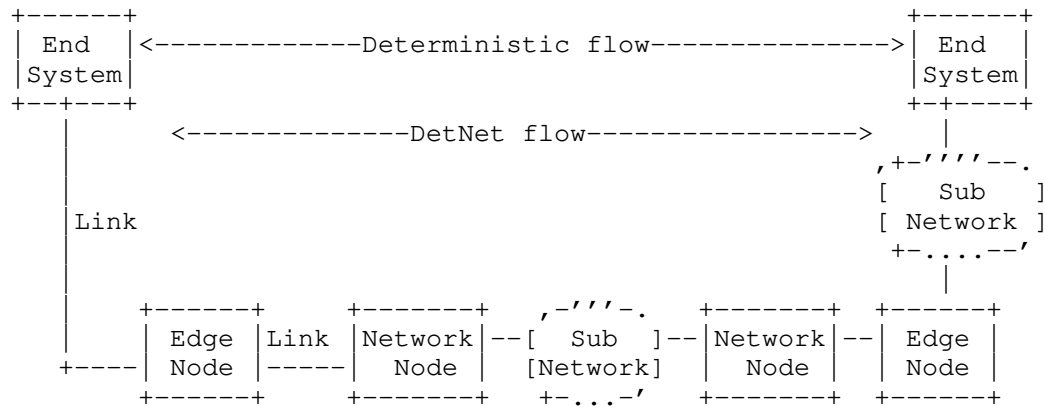


Figure 1: A Simple DetNet Enabled Network

This DetNet data plane is logically divided into two layers:

- o The DetNet Service layer provides adaptation of DetNet services. It is composed of a shim layer to carry DetNet flow specific attributes, which are needed during forwarding. End systems originate and terminate the DetNet Service layer and are peers at the DetNet Service layer.
- o The DetNet Transport layer is supported by all DetNet aware systems and nodes. It operates below the DetNet Service layer. The DetNet Transport layer is used to relay traffic end to end across a DetNet domain.

Distinguishing the function of these two DetNet data plane layers helps to explore and evaluate various combinations of the data plane solutions available. This separation of DetNet layers, while helpful, should not be considered as formal requirement. For example, some technologies may violate these strict layers and still be able to deliver a DetNet service.

A number of data plane technology candidates are discussed later in this document. They can be mapped to the two layers as shown in Figure 2.

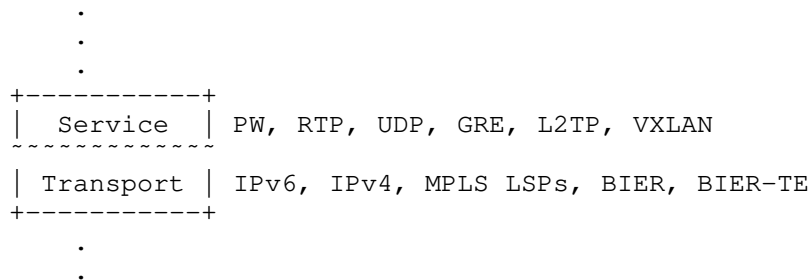


Figure 2: DetNet adaptation to data plane

Both the DetNet Service and the DetNet Transport layers are provided by a single technology in some solutions, e.g. the DetNet Service layer is PW and the DetNet Transport layer is MPLS in case of PW over MPLS. Nonetheless, both the DetNet Service and the DetNet Transport layers can include multiple technology layers in other solutions in order to provide the capabilities needed for DetNet flows. For instance, the DetNet Transport layer can comprise MPLS-in-GRE (Section 4.2.5) in one solution. In another solution, the DetNet Service layer can include, e.g., RTP in UDP (Section 4.2.8).

[Editor's note: I'm not sure if the remainder of this section says anything not present in the next section. Will need to revisit as part of the pre-pub review.]

There are many valid options to create a data plane solution for DetNet traffic by selecting a technology approach for the DetNet Service layer and also selecting a technology approach for the DetNet Transport layer. There are a high number of valid combinations. Therefore, not the combinations but the different technologies are evaluated along the criteria collected in Section 3. Different criteria apply for the DetNet Service layer and the DetNet Transport layer, however, some of the criteria are valid for both layers.

The criteria for the DetNet Service layer:

- #1 Encapsulation and overhead
- #2 Flow identification (Flow ID part of the DetNet flows)
- #3 Packet sequencing (sequence number)
- #5 Packet replication and deletion (note: only the packet deletion for seamless redundancy)
- #6 Operations, Administration and Maintenance (capabilities)
- #7 Time synchronization (e.g., time stamping)
- #8 Class and quality of service capabilities (DetNet Service specific)
- #10 Technical maturity

The criteria for the DetNet Transport layer:

- #1 Encapsulation and overhead
- #2 Flow identification
- #4 Explicit routes (network path)
- #5 Packet replication and deletion (note: only the packet replication for seamless redundancy)
- #6 Operations, Administration and Maintenance (capabilities)
- #7 Time synchronization (time/phase sync of nodes)
- #8 Class and quality of service capabilities (DetNet Transport specific)
- #9 Packet traceability (verification purposes)
- #10 Technical maturity

[Editor's note: #7 is more of OAM feature.]

Some of the criteria are relevant for both the DetNet Service and DetNet Transport layers. The two layers provide together what is needed to meet certain criteria, e.g., flow identification. Different aspects are valid for the two different layers for other criteria, e.g., time synchronization. Furthermore, technical maturity is a criteria valid for both layers.

### 3. Criteria for data plane solution alternatives

This section provides criteria to help to evaluate potential options. The criteria can be broken down based on layer. That is if the criteria is focused on delivering DetNet service adaptation, i.e., is concerned with the DetNet Service layer, or if the criteria is focused on transporting flows across an end to end DetNet domain.  
[Editor's note: which is TBD.]

Each deterministic networking data plane solution alternative is described and evaluated using the criteria described in this section. The used criteria enumerated in this section are selected so that they highlight the existence or lack of features that are expected or seen important to a solution alternative for the data plane solution.

#### 3.1. #? DetNet Service Interface

[Editor's note: this criteria needs a bit more discussion.]

One of the most fundamental differences between different potential data plane options is the basic addressing and headers used by DetNet clients. For example, is the basic service a Layer 2 (e.g., Ethernet) or Layer 3 (i.e., IP) service. This decision impacts how DetNet end points are addressed, and the basic forwarding logic of the DetNet Service layer.

### 3.2. #1 Encapsulation and overhead

Encapsulation and overhead is related to how the DetNet data plane carries DetNet user traffic. In several cases a deterministic flow has to be encapsulated inside other protocols, for example, when transporting a layer-2 Ethernet frame over an IP transport network. In some cases a former tunneling like encapsulation can be avoided by underlying transport protocol translation, for example, translating layer-2 Ethernet frame including addressing and flow identification into native IP traffic. Last it is possible that talkers and listeners handle deterministic flows natively in layer-3. This criteria concerns what is the encapsulation method the solution alternative support: tunneling like encapsulation, protocol translation or native layer-3 transport. In addition to the encapsulation mechanism this criteria is also concerned of the processing and specifically the encapsulate header overhead.

### 3.3. #2 Flow identification

The solution alternative has to provide means to identify specific deterministic flows. The flow identification can, for example, be explicit field in the data plane encapsulation header or implicitly encoded into the addressing scheme of the used data plane protocol or their combination. This criteria concerns the availability and details of deterministic flow identification the data plane protocol alternative has.

### 3.4. #3 Packet sequencing

[Editor's note: is in order delivery a strict requirement? if so, it should be stated as such and separately from any other requirement. There are multiple ways to solve this criteria.]

The solution alternative has to provide means for end systems to number packets sequentially and transport that sequencing information along with the sent packets. In addition to possible reordering packets one of the important uses for sequencing is detecting duplicates. In a case of intentional packet duplication a combination of flow identification and packet sequencing allows for detecting and discarding duplicates at the receiver (see Section 3.6 for more details). This criteria concerns the availability and details of the packet sequencing capabilities the data plane protocol alternative has.

### 3.5. #4 Explicit routes

The solution alternative has to provide a mechanism(s) for establishing explicit routes that all packets belonging to a deterministic flow will follow. The explicit route can be seen as a form of source routing or a pre-reserved path e.g., using some network management procedure. It should be noted that the explicit route does not need to be detailed to a level where every possible intermediate node along the path is part of the named explicit route. RSVP-TE [RFC3209] supports explicit routes, and typically provides pinned data paths for established LSPs. At Layer-2, the IEEE 802.1Qca [IEEE8021Qca] specification defines how to do explicit path control in a bridged network and its IETF counter part is defined in [I-D.ietf-isis-pcr]. This criteria concerns the available mechanisms for explicit routes for the data plane protocol alternative.

### 3.6. #5 Packet replication and deletion

The objective for supporting packet replication and deletion is to enable hitless (or lossless) 1+1 protection, which is also called Seamless redundancy in [I-D.finn-detnet-architecture]. Data plane solutions need to meet this objective independent of the particular solution used. In other words, a packet replication and deletion is one identified method for a data plane solution to provide seamless redundancy and other methods, if so identified, are permissible.

The solution alternative has to provide means for end systems and/or relay systems to be able to replicate packets, and later eliminate all but one of the replicas, at multiple points in the network in order to ensure that one (or more) equipment failure event(s) still leave at least one path intact for a deterministic networking flow. The goal is to enable hitless 1+1 protection in a way that no packet gets lost or there is no ramp up time when either one of the paths fails for one reason or another.

Another concern regarding packet replication is how to enforce replicated packets to take different route while the final destination still remains the same. With strict source routing, all the intermediate hops are listed and paths can be guaranteed to be non-overlapping. Loose source routing only signals some of the intermediate hops and it takes additional knowledge to ensure that there is no single point of failure.

[Editor's note: at the DetNet Transport layer this criteria does not concern packet deletion, only the packet replication. The packet deletion belongs to the DetNet Service layer]



The IEEE 802.1CB [IEEE8021CB] is an example of Ethernet-based solution that defines packet sequence numbering, packet replication, and duplicate packet identification and deletion. The deterministic networking data plane solution alternative at layer-3 has to provide equivalent functionality. This criteria concerns the available mechanisms for packet replication and duplicate deletion the data plane protocol alternative has.

### 3.7. #6 Operations, Administration and Maintenance

The solution alternative should demonstrate an availability of appropriate standardized OAM tools that can be extended for deterministic networking purposes with a reasonable effort, when required. The OAM tools do not necessarily need to be specific to the data plane protocol as it could be the case, for example, with MPLS-based data planes. But any OAM-related implications or requirements on data plane hardware must be considered.

### 3.8. #7 Time synchronization

Time synchronization between DetNet systems and nodes can be used to enable fine grain scheduling of traffic along an end-to-end data path. Such scheduling can be used to deliver very low jitter and latency. [DetNet-ARCH] refers to a synchronization target of less than 1 microsecond. Meeting such time synchronization objectives is likely to require specific hardware support, both at the synchronization protocol level and at the (time synchronized) packet scheduling level. It is worth noting that certain aspects of time synchronization and packet scheduling may be provided by the underlying sub-net technology, e.g., [IEEE802.1Qbv] and [IEEE802.1Qch].

### 3.9. #8 Class and quality of service capabilities

Class and quality of service, i.e., CoS and QoS, are terms that are often used interchangeably and confused. In the context of DetNet, CoS is used to refer to mechanisms that provide traffic forwarding treatment based on aggregate group basis and QoS is used to refer to mechanisms that provide traffic forwarding treatment based on a specific DetNet flow basis. Examples of CoS mechanisms include DiffServ which is enabled by IP header differentiated services code point (DSCP) field [RFC2474] and MPLS label traffic class field [RFC5462], and at Layer-2, by IEEE 802.1p priority code point (PCP).

Quality of Service (QoS) mechanisms for flow specific traffic treatment typically includes a guarantee/agreement for the service, and allocation of resources to support the service. Example QoS mechanisms include discrete resource allocation, admission control,

flow identification and isolation, and sometimes path control, traffic protection, shaping, policing and remarking. Example protocols that support QoS control include Resource ReSerVation Protocol [RFC2205] (RSVP) and RSVP-TE [RFC3209] and [RFC3473].

A critical DetNet service enabled by QoS (and perhaps CoS) is delivering zero congestion loss. There are different mechanisms that maybe used separately or in combination to deliver a zero congestion loss service. The key aspect of this objective is that DetNet packets are not discarded due to congestion at any point in a DetNet aware network.

In the context of the data plane solution there should be means for flow identification, which then can be used to map a flow against specific resources and treatment in a node enforcing the QoS. Hereto, certain aspects of CoS and QoS may be provided by the underlying sub-net technology, e.g., actual queuing or IEEE 802.3x priority flow control (PFC).

### 3.10. #9 Packet traceability

For the network management and specifically for tracing implementation or network configuration errors any means to find out whether a packet is a replica, which node performed replication, and which path was intended for the replica, can be very useful. This criteria concerns the availability of solutions for tracing packets in the context of data plane protocol alternative. Packet trace is a form of OAM.

### 3.11. #10 Technical maturity

The technical maturity of the data plane solution alternative is crucial, since it basically defines the effort, time line and risks involved for the use of the solution in deployments. For example, the maturity level can be categorized as available immediately, available with small extensions, available with repurposing/ redefining portions of the protocol or its header fields. Yet another important measure for maturity is the deployment experience. This criteria concerns the maturity of the data plane protocol alternative as the solution alternative. This criteria is particularly important given, as previously noted, that the DetNet data plane solution is expected to impact, i.e., be supported in, hardware.

#### 4. Data plane solution alternatives

The following sections describe and rate deterministic data plane solution alternatives. In "Analysis and Discussion" section each alternative is evaluated against the criteria given in Section 3 and rated using the following: (M)eets the criteria, (W)ork needed, and (N)ot suitable or too much work envisioned.

##### 4.1. DetNet Transport layer technologies

###### 4.1.1. Native IPv6 transport

###### 4.1.1.1. Solution description

This section investigates the application of native IPv6 [RFC2460] as the data plane for deterministic networking along the criteria collected in Section 3.

The application of higher OSI layer headers, i.e., headers deeper in the packet, can be considered. Two aspects have to be taken into account for such solutions. (i) Those header fields can be encrypted. (ii) Those header fields are deeper in the packet, therefore, routers have to apply deep packet inspection. See further details in Section 4.2.8.

###### 4.1.1.2. Analysis and Discussion

###### Encapsulation and overhead (M/W)

The DetNet Service layer encapsulated DetNet flows are assumed to be handled natively at layer-3 by IPv6 at the first place. The fixed header of an IPv6 packet is 40 bytes [RFC2460]. This overhead is bigger if any Extension Header is used, and a generic behaviour for host and forwarding nodes is specified in [RFC7045]. However, the exact overhead (Section 3.2) depends on what solution is actually used to provide DetNet features, e.g., explicit routing or seamless redundancy if any of these is applied.

IPv6 has two types of Extension Headers that are processed by intermediate routers between the source and the final destination and may be of interest for the data plane signaling, the Routing Header that is used to direct the traffic via intermediate routers in a strict or loose source routing way, and the Hop-by-Hop Options Header that carries optional information that must be examined by every node along a packet's delivery path. The Hop-by-Hop Options Header, when present, must immediately follow the IPv6 Header and it is not possible to limit its processing to the end points of Source Routed segments.

IPv6 also provides a Destination Options Header that is used to carry optional information to be examined only by a packet's destination node(s). The encoding of the options used in the Hop-by-Hop and in the Destination Options Header indicates the expected behavior when a processing IPv6 node does not recognize the Option Type, e.g. skip or drop; it should be noted that due to performance restrictions nodes may ignore the Hop-by-Hop Option Header, drop packets containing a Hop-by-Hop Option Header, or assign packets containing a Hop-by-Hop Option Header to a slow processing path [I-D.ietf-6man-rfc2460bis] (e.g. punt packets from hardware to software forwarding which is highly detrimental to the performance).

The creation of new Extension Headers that would need to be processed by intermediate nodes is strongly discouraged. In particular, new Extension Header(s) having hop-by-hop behavior must not be created or specified. New options for the existing Hop-by-Hop Header should not be created or specified unless no alternative solution is feasible [RFC6564].

#### Flow identification (M/W)

The 20-bit flow label field of the fixed IPv6 header is suitable to distinguish different deterministic flows. But guidance on the use of the flow label provided by [RFC6437] places restrictions on how the flow label can be used. In particular, labels should be chosen from an approximation to a discrete uniform distribution. Additionally, existing implementations generally do not open APIs to control the flow label from the upper layers.

Alternatively, the Flow identification could be transported in a new option in the Hop-by-Hop Options Header.

#### Explicit routes (W)

The general assumption is that a Software-Defined Networking (SDN) [RFC7426] based approach is applied to compute, establish and manage the explicit routes, leveraging Traffic Engineering (TE) extensions to routing protocols [RFC5305] [I-D.ietf-idr-ls-distribution] and evolving to the Path Computation Element (PCE) Architecture [RFC5440], though a number of issues remain to be solved [RFC7399].

Segment Routing (SR) [I-D.ietf-spring-segment-routing] is a new initiative to equip IPv6 with explicit routing capabilities. The idea for the DetNet data plane would be to apply SR to IPv6 with the addition of a new type of routing extension header

[I-D.ietf-6man-segment-routing-header] to explicitly signal the path in the data plane between the source and the destination, and/or between replication points and elimination points if this functionality is used.

Another concern regarding packet replication is how to enforce replicated packets to take different route while the final destination still remains the same. With strict source routing, all the intermediate hops are listed and paths can be guaranteed to be non-overlapping. Loose source routing only signals some of the intermediate hops and it takes additional knowledge to ensure that there is no single point of failure.

Packet replication (W) The functionality of replicating a packet exists in IPv6 but is limited to multicast flows.

In order to enforce replicated packets to take different routes, IP-in-IP encapsulation and Segment Routing could be leveraged to signal a segment in a packet. A replication point would insert a different routing header in each copy it makes, the routing header providing explicitly the hops to the elimination point for that particular replica of the packet, in a strict or in a loose source routing fashion. An elimination point would pop the routing headers from the various copies it gets and forward or receive the packet if it is the final destination.

Operations, Administration and Maintenance (M/W)

IPv6 enjoys the existing toolbox for generic IP network management. However, IPv6 specific management features are still not at the level of that IPv4 has. This specifically concerns the areas that are IPv6 specific, for example, related to neighbor discovery protocol (ND), stateless address autoconfiguration (SLAAC), subscriber identification, and security. While the standards are already mostly in place the implementations in deployed equipment can be lacking or inadequate for commercial deployments. This is largely only an issue with old existing equipment.

Class and quality of service capabilities (M)

The traffic class field of the fixed IPv6 header is designed for this purpose.

Packet traceability (M/W/N)

The traceability of replicated packets involves the capability to resolve which replication point issued a particular copy of a packet, which segment was intended for that replica, and which particular packet of which particular flow this is. Sequence also depends on the sequencing mechanism. As an example, the replication point may be indicated as the source of the packet if IP-in-IP encapsulation is used to forward along segments. Another alternate to IP-in-IP tunneling along segments would be to protect the original source address in a destination option similar to the Home Address option [RFC6275] and then use the address of the replication point as source in the IP header.

The traceability also involves the capability to determine if a particular segment is operational. While IPv6 as such has no support for reversing a path, it appears source route extensions such as the one defined for segment routing could be used for tracing purposes. Though it is not a usual practice, IPv6 [RFC2460] expects that a Source Route path may be reversed, and the standard insists that a node must not include the reverse of a Routing Header in the response unless the received Routing Header was authenticated.

#### Technical maturity (M/W)

IPv6 has been around about 20 years. However, large scale global and commercial IPv6 deployments are rather new dating only few years back to around 2012. While IPv6 has proven itself there are number of small issues to work on as they show up once operations experience grows.

The Cisco 6Lab site [1] provides information on IPv6 deployment per country, indicating figures for prefixes, transit AS, content and users. Per this site, many countries, including Canada, Brazil, the USA, Germany, France, Japan, Portugal, Sweden, Finland, Norway, Greece, and Ecuador, achieve a deployment ratio above 30 percent, and the overall adoption reported by Google Statistics [2] is now above 10 percent.

#### 4.1.1.3. Summary

TBD.

#### 4.1.2. Native IPv4 transport

#### 4.1.2.1. Solution description

IPv4 [RFC0791] is in principle the same as IPv6, except that it has a smaller address space. However, IPv6 was designed around the fact that extension headers are an integral part of the protocol and operation from the beginning, although the practice may some times prove differently [I-D.ietf-v6ops-ipv6-ehs-in-real-world]. IPv4 never really needed any extension headers to work properly, thus support for IPv4 options outside closed networks cannot typically be guaranteed. In the context of deterministic networking data plane solutions the major difference between IPv4 and IPv6 seems to be the practical support for header extensibility. Anything below and above the IP header independent of the version is practically the same.

#### 4.1.2.2. Analysis and Discussion

##### Encapsulation and overhead (M)

The fixed header of an IPv4 packet is 20 bytes [RFC0791]. IP options add overhead and the maximum IPv4 header size is 60 octets leaving only 40 octets for possible options.

##### Flow identification (W/N)

The IPv4 header has a 16-bit identification field that was originally intended for assisting fragmentation and reassembly of IPv4 packets as described in [RFC0791]. The identification field has also been proposed to be used for actually identifying flows between two IP addresses and a given protocol for detecting and removing duplicate packets [RFC1122]. However, recent update [RFC6864] to both [RFC0791] and [RFC1122] restricts the use of IPv4 identification field only to fragmentation purposes.

The IPv4 also has a stream identifier option [RFC0791], which contains a 16-bit SATNET stream identifier. However, the option has been deprecated [RFC6814]. The conclusion is that stream identification does not work nicely with IPv4 header alone and a traditional 5-tuple identification might not also be enough in a case of a flow duplication. For a working solution upper layer protocol headers such as the RTP are required for unambiguous flow identification.

##### Explicit routes (M/W)

IPv4 has two source routing option specified: the loose source and record route option (LSRR), and the strict source and record route option (SSRR) [RFC0791]. The support of these options in the

Internet is questionable but within a closed network the support may be assumed.

#### Packet replication (W/N)

The functionality of replicating a packet exists in IPv4 but is limited to multicast flows. In general the issue regarding the IPv6 packet replication also applies to IPv4. Duplicate packet detection for IPv4 is studied in [RFC6621] to a great detail in the context of simplified multicast forwarding.

#### Operations, Administration and Maintenance (M)

IPv4 enjoys the extensive and "complete" existing toolbox for generic IP network management.

#### Class and quality of service capabilities (M)

The type of service (TOS) field of the fixed IPv4 header is designed for this purpose.

#### Packet traceability (W/N)

The IPv4 has a traceroute option [RFC1393] that could be used to record the route the packet took. However, the option has been deprecated [RFC6814]. Similarly to IPv6 new work would be needed to allow better traceability of IPv4 packets.

#### Technical maturity (M/W)

IPv4 can be considered mature technology with over 30 years of implementation, deployment and operations experience. However, no new IPv4 standards development is "allowed" anymore [RFC6540][I-D.ietf-sunset4-gapanalysis].

#### 4.1.2.3. Summary

TBD.

#### 4.1.3. Multiprotocol Label Switching (MPLS)

##### 4.1.3.1. Solution description

Multiprotocol Label Switching Architecture (MPLS) [RFC3031] and its variants, MPLS with Traffic Engineering (MPLS-TE) [RFC3209] and [RFC3473], and MPLS Transport Profile (MPLS-TP) [RFC5921] is a widely deployed technology that switches traffic based on MPLS label stacks



[RFC3032] and [RFC5960]. MPLS is the foundation for Pseudowire-based services Section 4.2.5 and emerging technologies such as Bit-Indexed Explicit Replication (BIER) Section 4.2.7.1 and Source Packet Routing [3].

MPLS supports the equivalent of both the DetNet Service and DetNet Transport layers, and provides a very rich set of mechanisms that can be reused directly, and perhaps augmented in certain cases, to deliver DetNet services. At the DetNet Transport layer, MPLS provides forwarding, protection and OAM services. At the DetNet Service Layer it provides client service adaption, directly, via Pseudowires Section 4.2.5 and via other label-like mechanisms such as EPVN Section 4.2.6. A representation of these options are shown in Figure 3.

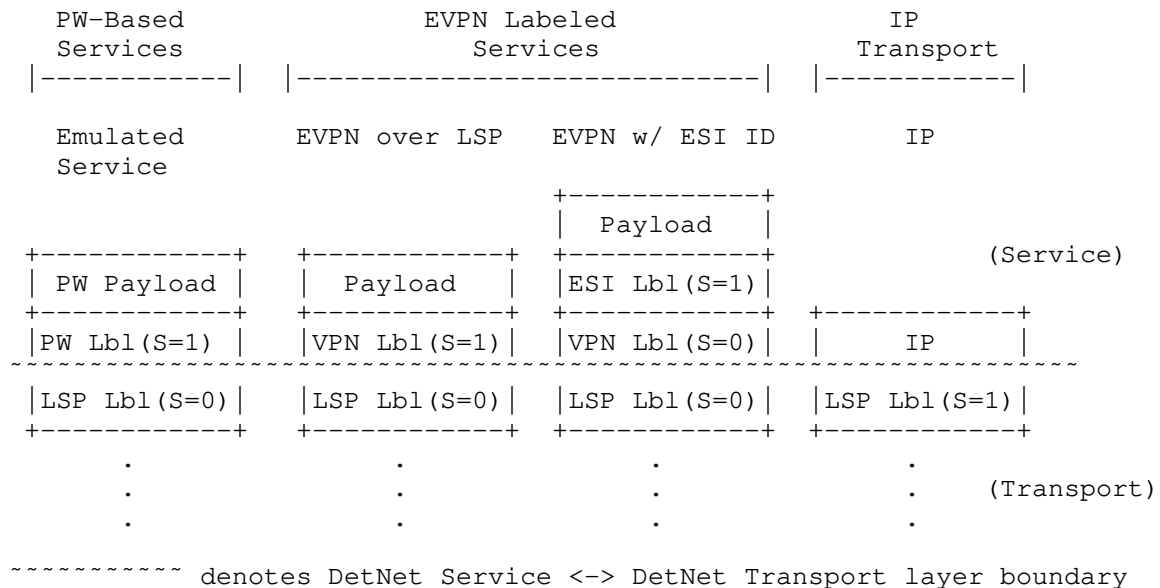


Figure 3: MPLS-based Services

MPLS can be controlled in a number of ways including via a control plane, via the management plane, or via centralized controller (SDN) based approaches. MPLS also provides standard control plane reference points. Additional information on MPLS architecture and control can be found in [RFC5921]. A summary of MPLS control plane related functions can be found in [RFC6373]. The remainder of this section will focus [RFC6373]. The remainder of this section will focus on the MPLS transport data plane, additional information on the MPLS service data plane can be found below in Section 4.2.4.

The following is a work in progress and draws heavily from [RFC5960] and may be updated, replaced or removed.

Encapsulation and forwarding of packets traversing MPLS LSPs follows standard MPLS packet encapsulation and forwarding as defined in [RFC3031], [RFC3032], [RFC5331], and [RFC5332].

Data plane Quality of Service capabilities are included in the MPLS in the form of Traffic Engineered (TE) LSPs [RFC3209] and the MPLS Differentiated Services (DiffServ) architecture [RFC3270]. Both E-LSP and L-LSP MPLS DiffServ modes are defined. The Traffic Class field (formerly the EXP field) of an MPLS label follows the definition of [RFC5462] and [RFC3270].

Except for transient packet reordering that may occur, for example, during fault conditions, packets are delivered in order on L-LSPs, and on E-LSPs within a specific ordered aggregate.

The Uniform, Pipe, and Short Pipe DiffServ tunneling and TTL processing models are described in [RFC3270] and [RFC3443] and may be used for MPLS LSPs.

Equal-Cost Multi-Path (ECMP) load-balancing is possible with MPLS LSPs and can be avoided using a number of techniques. The same holds for Penultimate Hop Popping (PHP).

MPLS includes the following LSP types:

- o Point-to-point unidirectional
- o Point-to-point associated bidirectional
- o Point-to-point co-routed bidirectional
- o Point-to-multipoint unidirectional

Point-to-point unidirectional LSPs are supported by the basic MPLS architecture [RFC3031].

A point-to-point associated bidirectional LSP between LSRs A and B consists of two unidirectional point-to-point LSPs, one from A to B and the other from B to A, which are regarded as a pair providing a single logical bidirectional transport path.

A point-to-point co-routed bidirectional LSP is a point-to-point associated bidirectional LSP with the additional constraint that its two unidirectional component LSPs in each direction follow the same path (in terms of both nodes and links). An important property of

co-routed bidirectional LSPs is that their unidirectional component LSPs share fate.

A point-to-multipoint unidirectional LSP functions in the same manner in the data plane, with respect to basic label processing and packet-switching operations, as a point-to-point unidirectional LSP, with one difference: an LSR may have more than one (egress interface, outgoing label) pair associated with the LSP, and any packet it transmits on the LSP is transmitted out all associated egress interfaces. Point-to-multipoint LSPs are described in [RFC4875] and [RFC5332]. TTL processing and exception handling for point-to-multipoint LSPs is the same as for point-to-point LSPs.

Additional data plane capabilities include Linear Protection, [RFC6378] and [RFC7271]. And the in progress work on MPLS support for time synchronization [I-D.ietf-mpls-residence-time].

#### 4.1.3.2. Analysis and Discussion

##### #? DetNet Service Interface (M)

The DetNet service interface is enabled through the DetNet Service Layer it provides client service adaption, directly, via Pseudowires Section 4.2.5 and via other label-like mechanisms such as EPVN Section 4.2.6.

##### #1 Encapsulation and overhead (M)

There are two perspectives to consider when looking at encapsulation. The first is encapsulation to support services. These considerations are part of the DetNet service layer and are covered below, see Sections 4.2.5 and 4.2.6.

The second perspective relates to encapsulation, if any, is needed to transport packets across network. In this case, the MPLS label stack, [RFC3032] is used to identify flows across a network. MPLS labels are compact and highly flexible. They can be stacked to support client adaptation, protection, network layering, source routing, etc.

##### #2 Flow identification (M)

MPLS label stacks provide highly flexible ways to identify flows. Basically, they enable the complete separation of traffic classification from traffic treatment and thereby enable arbitrary combinations of both.

## #3 Packet sequencing (M)

Packet ordering in MPLS is generally similar to packet ordering in Ethernet. MPLS implementations can also support ECMP for certain types of traffic which can lead to out of order delivery. There are defined techniques to avoid ECMP and ensure in order delivery during normal operation. Out of order delivery is still possible in certain MPLS protection scenarios. If additional ordering mechanisms are required, these are likely to be implemented at the DetNet Service Layer.

## #4 Explicit routes (M)

MPLS supports explicit routes based on how LSPs are established, e.g., via TE explicit routes [RFC3209]. Additional, but not required, additional capabilities are being defined as part of Segment Routing (SR) [I-D.ietf-spring-segment-routing].

## #5 Packet replication and deletion (M/W)

At the MPLS LSP level, there are mechanisms defined to provide 1+1 protection. The current definitions [RFC6378] and [RFC7271] use OAM mechanisms to support and coordinate protection switching and packet loss is possible during a switch. While such this level of protection may be sufficient for many DetNet applications, when truly hitless (i.e., zero loss) switching is required additional mechanisms will be needed. It is expected that these additional mechanisms will be defined at the DetNet Service Layer.

## #6 Operations, Administration and Maintenance (M)

MPLS already includes a rich set of OAM functions at both the Service and Transport Layers. This includes LSP ping [ref] and those enabled via the MPLS Generic Associated Channel [RFC5586] and registered by IANA [4].

## #7 Time synchronization (M/W)

MPLS itself does not provide any time synchronization service. The expectation is that the actual time-based scheduling will be provided by the sub-network layer, e.g., by [TSNTG], and that the DetNet transport layer will merely need to facilitate time synchronization (with hardware support) across multiple sub-network domains and technologies. Work is in progress

[I-D.ietf-mpls-residence-time] that may satisfy, or serve as a building block for, DetNet time synchronization.

#### #8 Class and quality of service capabilities (M/W)

As previously mentioned, Data plane Quality of Service capabilities are included in the MPLS in the form of Traffic Engineered (TE) LSPs [RFC3209] and the MPLS Differentiated Services (DiffServ) architecture [RFC3270]. Both E-LSP and L-LSP MPLS DiffServ modes are defined. The Traffic Class field (formerly the EXP field) of an MPLS label follows the definition of [RFC5462] and [RFC3270]. One potential open area of work is synchronized, time based scheduling.

#### #9 Packet traceability (M)

MPLS supports multiple tracing mechanisms. A control based one is defined in [RFC3209]. An OAM based mechanism is defined in MPLS On-Demand Connectivity Verification and Route Tracing [RFC6426].

#### #10 Technical maturity (M)

MPLS as a mature technology that has been widely deployed in many networks for many years. Numerous vendor products and multiple generations of MPLS hardware have been built and deployed.

#### 4.1.3.3. Summary

MPLS is a mature technology that has been widely deployed. Numerous vendor products and multiple generations of MPLS hardware have been built and deployed. MPLS LSPs support a significant portion of the identified DetNet data plane criteria today. Aspects of the DetNet data plane that are not fully supported can be incrementally added.

#### 4.2. DetNet Service layer technologies

##### 4.2.1. Generic Routing Encapsulation (GRE)

###### 4.2.1.1. Solution description

Generic Routing Encapsulation (GRE) [RFC2784] provides an encapsulation of an arbitrary network layer protocol over another

arbitrary network layer protocol. The encapsulation of a GRE packet can be found in Figure 4.

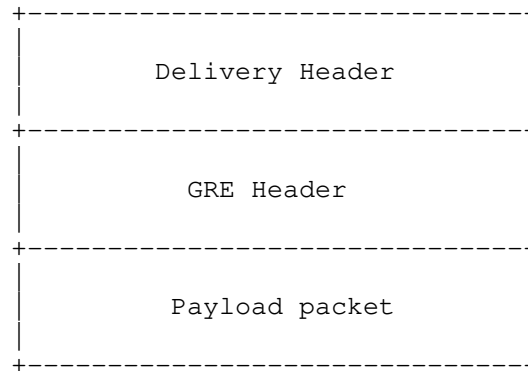


Figure 4: Encapsulation of a GRE packet

Based on RFC2784, [RFC2890] further includes sequencing number and Key in optional fields of the GRE header, which may help to transport DetNet traffic flows over IP networks. The format of a GRE header is presented in Figure 5.

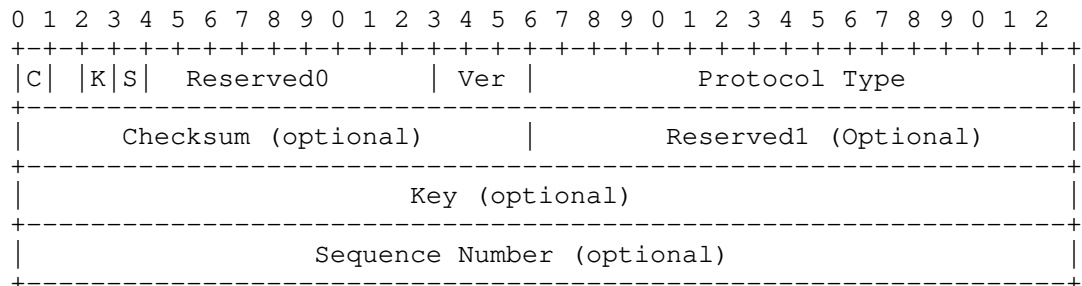


Figure 5: Format of a GRE header

#### 4.2.1.2. Analysis and Discussion

##### Encapsulation and overhead (M)

GRE provides encapsulation for a network layer protocol over another network layer protocol. A new protocol type for DetNet traffics should be allocated as an "Ether Type" in [RFC1700] and in IANA Ethernet Numbers. [5] The fixed header of a GRE packet is 4 octets while the maximum header is 16 octets with optional fields in Figure 5.

## Flow identification (W)

There is no flow identification field in GRE header. However, it can rely on the flow identification mechanism applied in the delivery protocols, such as flow identification stated in IP Sections 4.1.1 and 4.1.2 when the delivery protocols are IPv6 and IPv4 respectively. Alternatively, the Key field can also be extended to carry the flow identification. The size of Key field is 4 octets.

## Packet sequencing (M)

As stated in Section 4.2.1, GRE provides an optional sequencing number in its header to provide sequencing services for packets. The size of the sequencing number is 32 bits.

## Duplicate packet deletion (N)

GRE has no packet replication and deletion currently in its header and should be extended or rely on delivery protocols.

## Operations, Administration and Maintenance (W/N)

[note: rely on the delivery protocol] GRE has no packet replication and deletion currently and should be relied on delivery protocols.

## Time synchronization (W/N)

[note: rely on the delivery protocol] GRE has no packet replication and deletion currently and should be relied on delivery protocols.

## Class and quality of service capabilities (W/N)

[note: rely on the delivery protocol] GRE has no packet replication and deletion currently and should be relied on delivery protocols. For the class of service capability, an optional code point field to indicate CoS of a traffic can be extended in GRE header.

## Technical maturity (M)

GRE has been developed over 20 years. The delivery protocol mostly used is IPv4, while the IPv6 support for GRE is to be standardized now in IETF as [I-D.ietf-intarea-gre-ipv6]. Due to its good extensibility, GRE is also extended to support network virtualization in Data Center, which is NVGRE [RFC7637].

#### 4.2.1.3. Summary

TBD.

#### 4.2.2. Layer-2 Tunneling Protocol (L2TP)

[Editor's note: L2TPv3 [RFC3931] content to be provided later, if needed]

#### 4.2.3. Virtual Extensible LAN (VXLAN)

[Editor's note: VXLAN [RFC7348] content to be provided later, if needed]

#### 4.2.4. MPLS-based Services

##### 4.2.4.1. Solution description

MPLS supports the equivalent of both the DetNet Service and DetNet Transport layers. This, as well as a general overview of MPLS, is covered above in Section 4.1.3. This section will focus on the DetNet Service Layer it provides client service adaption, via Pseudowires in Section 4.2.5 and via native and other label-like mechanisms such as EPVN in Section 4.2.6. A representation of these options was previously discussed and is shown in Figure 3.

##### 4.2.4.2. Analysis and Discussion

#? DetNet Service Interface (M)

The following text is adapted from [RFC5921]:

The MPLS native service adaptation functions interface the client layer network service to MPLS. For Pseudowires, these adaptation functions are the payload encapsulation described in Section 4.4 of [RFC3985] and Section 6 of [RFC5659]. For network layer client services, the adaptation function uses the MPLS encapsulation format as defined in [RFC3032].

The purpose of this encapsulation is to abstract the data plane of the client layer network from the MPLS data plane, thus contributing to the independent operation of the MPLS network.

MPLS may itself be a client of an underlying server layer. MPLS can thus also be bounded by a set of adaptation functions to this server layer network, which may itself be MPLS. These adaptation functions provide encapsulation of the MPLS frames and for the



transparent transport of those frames over the server layer network.

While MPLS service can be provided on an end-system to end-system basis, it's more likely that DetNet service will be provided over Pseudowires as described in Section 4.2.5 or via an EPVN-based service described in Section 4.2.6 .

#### #1 Encapsulation and overhead (M)

MPLS labels in the label stack may be used to identify transport paths, see Section 4.1.3, or as service identifiers. Typically a single label is used for service identification. Additional details on how client adaptation makes use of such labels is part of actual client-related adaptation processing, see Sections 4.2.5 and 4.2.6.

#### #2 Flow identification (M)

This is basically the same as MPLS at the DetNet transport layer. MPLS label stacks provide highly flexible ways to identify flows. Basically, they enable the complete separation of traffic classification from traffic treatment and thereby enable arbitrary combinations of both. Typically a separate label will be added per service being supported by a node.

#### #3 Packet sequencing (M)

This is the same as MPLS at the DetNet transport layer. If additional ordering mechanisms are required, these will be needed (and added) in client-related adaptation processing, see Sections 4.2.5 and 4.2.6.

#### #4 Explicit routes (N/A)

Explicit routes are part of the DetNet transport layer, see Section 4.2.6, or as part of multi-segment PWEs, Section 4.2.5.

#### #5 Packet replication and deletion (M/W)

This is the same as MPLS at the DetNet transport layer. Additional capability may also be provided as part of client-related adaptation processing see Section 4.2.5.

## #6 Operations, Administration and Maintenance (M)

This is the same as MPLS at the DetNet transport layer. Additional capability may also be provided as part of client-related adaptation processing.

## #7 Time synchronization (TBD)

It's unclear at this time if any additional capability is needed at this level.

## #8 Class and quality of service capabilities (M/W)

The MPLS client inherits its Quality of Service (QoS) from the MPLS transport layer, which in turn inherits its QoS from the server (sub-network) layer. The server layer therefore needs to provide the necessary QoS to ensure that the MPLS client QoS commitments can be satisfied.

## #9 Packet traceability (M)

This is the same as MPLS at the DetNet transport layer.

## #10 Technical maturity (M)

This is the same as MPLS at the DetNet transport layer.

## 4.2.4.3. Summary

This is the same as MPLS at the DetNet transport layer. MPLS is a mature technology that has been widely deployed. Numerous vendor products and multiple generations of MPLS hardware have been built and deployed. MPLS LSPs support a significant portion of the identified DetNet data plane criteria today. Aspects of the DetNet data plane that are not fully supported can be incrementally added.

## 4.2.5. Pseudo Wire Emulation Edge-to-Edge (PWE3)

## 4.2.5.1. Solution description

Pseudo Wire Emulation Edge-to-Edge (PWE3) [RFC3985] or simply PseudoWires (PW) provide means of emulating the essential attributes and behaviour of a telecommunications service over a packet switched network (PSN) using IP or MPLS transport. In addition to traditional telecommunications services such as T1 line or Frame Relay, PWs also provide transport for Ethernet service [RFC4448] and for generic packet service [RFC6658]. Figure 6 illustrate the reference PWE3 stack model.

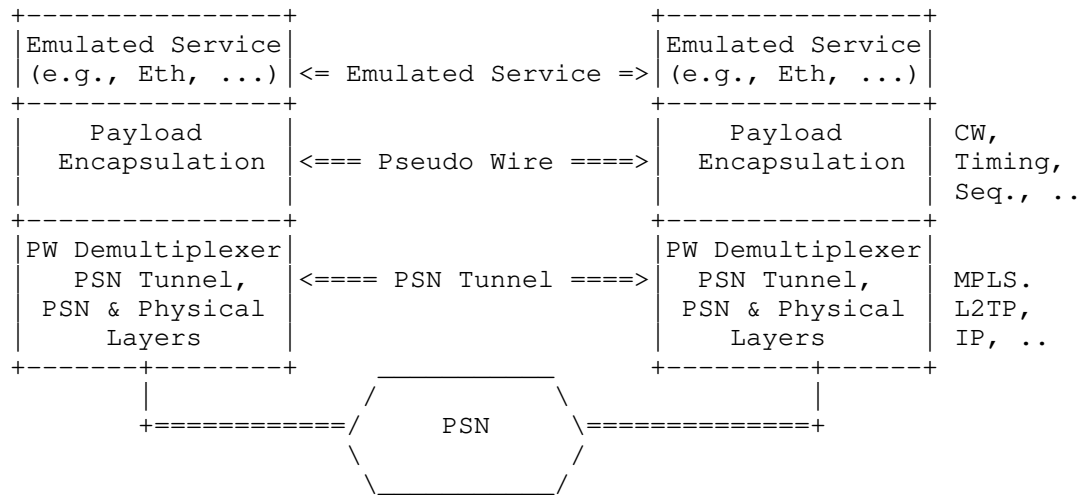


Figure 6: PWE3 protocol stack reference model

PWs appear as a good data plane solution alternative for a number of reasons. PWs are a proven and deployed technology with a rich OAM control plane [RFC4447], and enjoy the toolbox developed for MPLS networks. Furthermore, PWs may have an optional Control Word (CW) as part of the payload encapsulation between the PSN and the emulated service that is, for example, capable of frame sequencing and duplicate detection. The encapsulation layer may also provide timing [RFC5087].

PWs can be also used if the PSN is IP, which enables the application of PWs in networks that do not have MPLS enabled in their core routers. One approach to provide PWs over IP is to provide MPLS over IP in some way and then leverage what is available for PWs over MPLS. The following standard solutions are available both for IPv4 and IPv6 to follow this approach. The different solutions have different overhead as discussed in the following subsection. The MPLS-in-IP

encapsulation is specified by [RFC4023]. The IPv4 Protocol Number field or the IPv6 Next Header field is set to 137, which indicates an MPLS unicast packet. (The use of the MPLS-in-IP encapsulation for MPLS multicast packets is not supported.) The MPLS-in-GRE encapsulation is specified in [RFC4023], where the IP header (either IPv4 or IPv6) is followed by a GRE header, which is followed by an MPLS label stack. The protocol type field in the GRE header is set to MPLS Unicast (0x8847) or Multicast (0x8848). MPLS over L2TPv3 over IP encapsulation is specified by [RFC4817]. The MPLS-in-UDP encapsulation is specified by [RFC7510], where the UDP Destination Port indicates tunneled MPLS packet and the UDP Source Port is an entropy value that is generated by the encapsulator to uniquely identify a flow. MPLS-in-UDP encapsulation can be applied to enable UDP-based ECMP (Equal-Cost Multipath) or Link Aggregation. All these solutions can be secured with IPsec.

#### 4.2.5.2. Analysis and Discussion

##### Encapsulation and overhead (M)

PWs offer encapsulation services practically for any types of payloads over any PSN. New PW types need a code point allocation [RFC4446] and in some cases an emulated service specific document.

Specifically in the case of the MPLS PSN the PW encapsulation overhead is minimal. Typically minimum two labels and a CW is needed, which totals to 12 octets. PW type specific handling might, however, allow optimizations on the emulated service in the provider edge (PE) device's native service processing (NSP) / forwarder function. These optimizations could be used, for example, to reduce header overhead. Ethernet PWs already have rather low overhead [RFC4448]. Without a CW and VLAN tags the Ethernet header gets reduced to 14 octets (minimum Ethernet header overhead is 26).

The overhead is somewhat bigger in case of IP PSN if an MPLS over IP solution is applied to provide PWs. IP adds at least 20 (IPv4) or 40 (IPv6) bytes overhead to the PW over MPLS overhead; furthermore, the GRE, L2TPv3, or UDP header has to be taken into account if any of these further encapsulations is used.

##### Flow identification (M)

[Editor's note: this criteria has not been checked against the latest view of flow identification after the separation of transport and service layers.]

PWs provide multiple layers of flow identification, especially in the case of the MPLS PSN. The PWs are typically prepended with a PW label that can be used to identify a specific PW. Furthermore, the PSN also uses one or more labels to transport packets over a specific label switched paths (that then would carry PWs). IP (and other) PSNs may need other mechanisms, such as, UDP port numbers, upper layer protocol header (like RTP) or some IP extension header to provide required flow identification.

#### Packet sequencing (M)

As mentioned earlier PWs may contain an optional CW that is able to provide sequencing services. The size of the sequence number in the generic CW is 16 bits, which might be, depending on the used link and DetNet flow speed be too little.

#### Duplicate packet deletion (W)

The PW duplicate detection mechanism also exists in theory [RFC3985] but no emulated service makes use of it currently.

#### Operations, Administration and Maintenance (M/W)

PWs have rich control plane for OAM and in a case of the MPLS PSN enjoy the full control plane toolbox developed for MPLS network OAM likewise IP PSN have the full toolbox of IP network OAM tools. There could be, however, need for deterministic networking specific extensions for the mentioned control planes.

#### Time synchronization (M/W)

It is possible to carry time synchronization information as part of the PW encapsulation layer (see for example [RFC5087]). Whether the timing precision is enough for all deterministic networking use cases vary, and it is possible existing mechanisms are not adequate for all use cases. IP PSNs have already demonstrated the use of time synchronization as a part of PWE3 [RFC5086].

#### Class and quality of service capabilities (M)

In a case of IP PSN the 6-bit differentiated services code point (DSCP) field can be used for indicating the class of service [RFC2474] and 2-bit field reserved for the explicit congestion notification (ECN) [RFC3168]. Similarly, in a case of MPLS PSN, there are 3-bit traffic class field (TC) [RFC5462] in the label reserved for for both Explicitly TC-encoded-PSC LSPs (E-LSP) [RFC3270] and ECN [RFC5129]. Due to the limited number of bits in

the TC field, their use for QoS and ECN functions restricted and intended to be flexible. Although the QoS/CoS mechanism is already in place some clarifications may be required in the context of deterministic networking flows, for example, if some specific mapping between bit fields have to be done.

#### Technical maturity (M)

PWs, IP and MPLS are proven technologies with wide variety of deployments and years of operational experience. Furthermore, the estimated work for missing functionality (packet replication and deletion) does not appear to be extensive, since the existing protection mechanism already get close to what is needed from the deterministic networking data plane solution.

#### 4.2.5.3. Summary

PseudoWires appear to be a strong candidate as the deterministic networking data plane solution alternative for the DetNet Service layer. The strong points are the technical maturity and the extensive control plane for OAM. This holds specifically for MPLS-based PSN.

Extensions are required to realize the packet replication and duplicate detection features of the deterministic networking data plane.

#### 4.2.6. MPLS-Based Ethernet VPN (EVPN)

##### 4.2.6.1. Solution description

MPLS-Based Ethernet VPN (EVPN), in the form documented in [RFC7432] and [RFC7209], is an increasingly popular approach to delivering MPLS-based Ethernet services and is designed to be the successor to Virtual Private LAN Service (VPLS), [RFC4664].

EVPN provides client adaptation and reuses the MPLS data plane discussed above in Section 4.2.4. In certain special cases, it also uses the PW MPLS Control Word. EVPN control is via BGP, [RFC7432], and may use TE-LSPs, e.g., controlled via [RFC3209] for MPLS transport. Additional EVPN related RFCs and in progress drafts are being developed by the BGP Enabled Services Working Group [6].

#### 4.2.6.2. Analysis and Discussion

##### #? DetNet Service Interface (M/W)

The service supported by EVPN is a layer 2 Ethernet virtual private network. While EVPN is typically envisioned to be deployed on provider edge systems, it is also possible to extent the EVPN service to a DetNet end or edge system if such service is needed.

##### #1 Encapsulation and overhead (M)

EVPN generally uses a single MPLS label stack entry to support its client adaptation service. The optional addition of a second label is also supported. In certain cases PW Control Word may also be used.

##### #2 Flow identification (W)

EVPN currently uses labels to identify flows per {Ethernet Segment Identifier, VLAN} or per MAC level. Additional definition will be needed to standardize identification of finer granularity DetNet flows.

##### #3 Packet sequencing (M/W)

Like MPLS, EVPN generally orders packets similar to Ethernet. Reordering is possible primarily during path changes and protection switching. In order to avoid misordering due to ECMP, EVPN uses the "Preferred PW MPLS Control Word" [RFC4385] or the entropy labels [RFC6790].

If additional ordering mechanisms are required, such mechanisms will need to be defined.

##### #4 Explicit routes (M)

EVPN itself doesn't offer support for explicit routes as it is simply an adaptation function. Explicit routes for EVPN at the DetNet transport layer would be provided via MPLS.

##### #5 Packet replication and deletion (M/W)

EVPN relies on the MPLS layer for all protection functions. See Section 4.1.3 and Section 4.2.4. Some extensions, either at the EVPN or MPLS levels, will be need to support those DetNet applications which require true hitless (i.e., zero loss) 1+1 protection switching. (Network coding may be an interesting alternative to investigate to delivering such hitless loss protection capability.)

#### #6 Operations, Administration and Maintenance (M/W)

Nodes supporting EVPN may participate in either or both Ethernet level and MPLS level OAM. It is likely that it may make sense to map or adapt the OAM functions at the different levels, but such has yet to be defined. [RFC6371] provides some useful background on this topic.

#### #7 Time synchronization (W)

The interface to the DetNet time synchronization service is still to be determined. If the service is accessed by end systems via IEEE defined mechanisms, then those mechanisms will need to be mapped to the MPLS provided mechanisms discussed in Section 4.1.3.

#### #8 Class and quality of service capabilities (M/W)

EVPN is largely silent on the topics of CoS and QoS, but the existing Ethernet and MPLS mechanisms can be directly used. While an implementation may support such mappings today, standardized mappings do not (yet) exist.

#### #9 Packet traceability (M)

EVPN nodes can utilize MPLS layers tracing mechanisms.

#### #10 Technical maturity

EVPN is a second (or third) generation MPLS-based L2VPN service standard. From a data plane standpoint it makes uses of existing MPLS data plane mechanisms. The mechanisms have been widely implemented and deployed.



#### 4.2.6.3. Summary

EVPN is the emerging successor to VPLS. EVPN is standardized, implemented and deployed. It makes use of the mature MPLS data plane. While offering a mature and very comprehensive set of features, certain DetNet required features are not fully/directly supported and additional standardization in these areas are needed. Examples include: mapping CoS and QoS; use of labels per DetNet flow, and hitless 1+1 protection.

#### 4.2.7. Bit Indexed Explicit Replication (BIER)

Bit Indexed Explicit Replication [I-D.ietf-bier-architecture] (BIER) is a network plane replication technique that was initially intended as a new method for multicast distribution. In a nutshell, a BIER header includes a bitmap that explicitly signals the listeners that are intended for a particular packet, which means that 1) the sender is aware of the individual listeners and 2) the BIER control plane is a simple extension of the unicast routing as opposed to a dedicated multicast data plane, which represents a considerable reduction in OPEX. For this reason, the technology faces a lot of traction from Service Providers. Section 4.2.7.1 discusses the applicability of BIER for replication in the DetNet.

The simplicity of the BIER technology makes it very versatile as a network plane signaling protocol. Already, a new Traffic Engineering variation is emerging that uses bits to signal segments along a TE path. While the more classical BIER is mainly a multicast technology that typically leverages a unicast distributed control plane through IGP extensions, BIER-TE is mainly a unicast technology that leverages a central computation to setup path, compute segments and install the mapping in the intermediate nodes. Section 4.2.7.2 discusses the applicability of BIER-TE for replication, traceability and OAM operations in DetNet.

##### 4.2.7.1. Base BIER

Bit-Indexed Explicit Replication (BIER) layer may be considered to be included into Deterministic Networking data plane solution. Encapsulation of a BIER packet in MPLS network presented in Figure 7

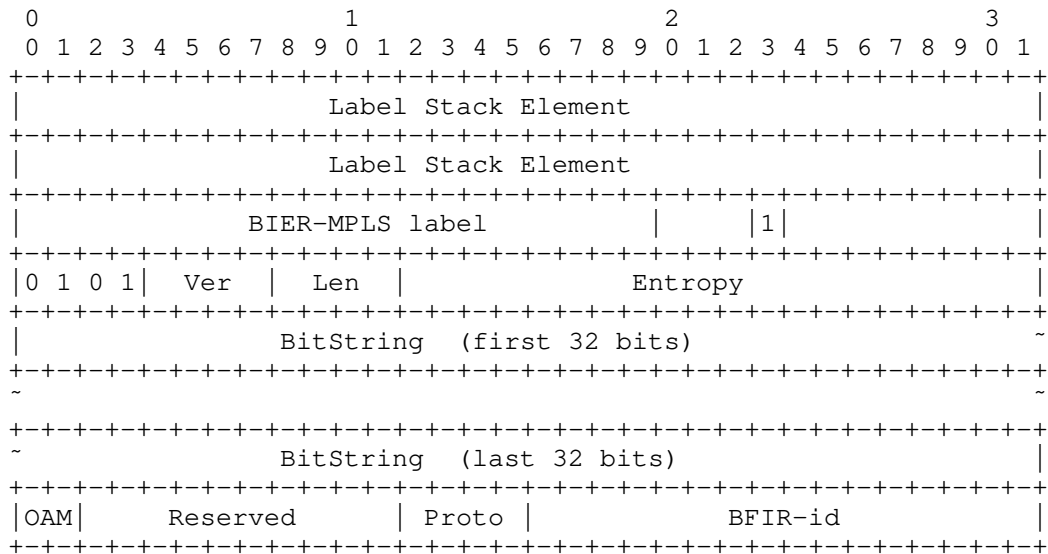


Figure 7: BIER packet in MPLS encapsulation

## 4.2.7.1.1. Solution description

The DetNet may be presented in BIER as distinctive payload type with its own Proto(col) ID. Then it is likely that DetNet will have the header that would identify:

- o Version;
- o Sequence Number;
- o Timestamp;
- o Payload type, e.g. data vs. OAM.

DetNet node, collocated with BFIR, may use multiple BIER sub-domains to create replicated flows. Downstream DetNet nodes, collocated with BFER, would terminate redundant flows based on Sequence Number and/or Timestamp information. Such DetNet may be BFER in one BIER sub-domain and BFIR in another. Thus DetNet flow would traverse several BIER sub-domains.

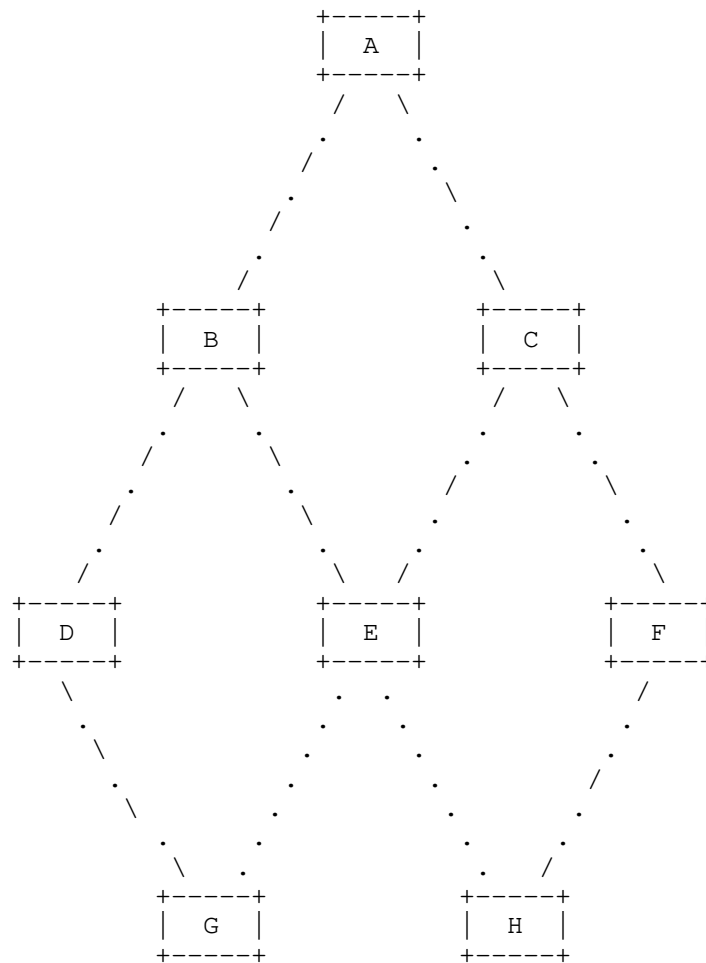


Figure 8: DetNet in BIER domain

Consider DetNet flow that must traverse BIER enabled domain from A to G and H. DetNet may use three BIER subdomains:

- o A-B-D-E-G (dash-dot): A is BFIR, E and G are BFERs,
- o A-C-E-F-H (dash-double-dot): A is BFIR, E and H are BFERs,
- o E-G-H (dotted): E is BFIR, G and H are BFERs.

DetNet node A sends DetNet into red and purple BIER sub-domains. DetNet node E receives DetNet packet and sends into green sub-domain while terminating duplicates and those that deemed too-late.

DetNet nodes G and H receive DetNet flows, terminate duplicates and those that are too-late.

#### 4.2.7.1.2. Analysis and Discussion

#### 4.2.7.1.3. Summary

#### 4.2.7.2. BIER - Traffic Engineering

An alternate use of Bit-Indexed Explicit Replication (BIER) uses bits in the BitString to represent adjacencies as opposed to destinations, as discussed in BIER Traffic Engineering (TE) [I-D.eckert-bier-te-arch].

The proposed function of BIER-TE in the DetNet data plane is to control the process of replication and elimination, as opposed to the identification of the flows or and the sequencing of packets within a flow.

At the path ingress, BIER-TE identifies the adjacencies that are activated for this packet (under the rule of the controller). At the egress, BIER-TE is used to identify the adjacencies where transmission failed. This information is passed to the controller, which in turn can modify the active adjacencies for the next packets.

The value is that the replication can be controlled and monitored with the granularity of a packet and a adjacency in a control loops that involves an external controller.

##### 4.2.7.2.1. Solution description

BIER-TE enables to activate the replication and elimination functions in a manner that is abstract to the data plane forwarding information. An adjacency, which is represented by a bit in the BIER header, can correspond in the data plane to an Ethernet hop, a Label Switched Path, or it can correspond to an IPv6 loose or strict source routed path.

In a nutshell, BIER-TE is used as follows:

- o A controller computes a complex path, sometimes called a track, which takes the general form of a ladder. The steps and the side rails between them are the adjacencies that can be activated on demand on a per-packet basis using bits in the BIER header.

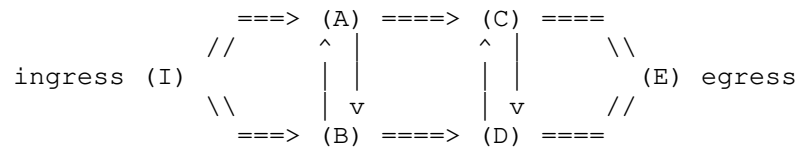


Figure 9: Ladder Shape with Replication and Elimination Points

- o The controller assigns a BIER domain, and inside that domain, assigns bits to the adjacencies. The controller assigns each bit to a replication node that sends towards the adjacency, for instance the ingress router into a segment that will insert a routing header in the packet. A single bit may be used for a step in the ladder, indicating the other end of the step in both directions.

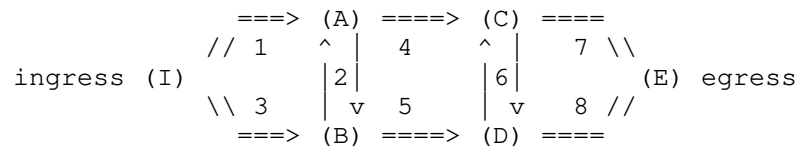


Figure 10: Assigning Bits

- o The controller activates the replication by deciding the setting of the bits associated with the adjacencies. This decision can be modified at any time, but takes the latency of a controller round trip to effectively take place. Below is an example that uses Replication and Elimination to protect the A->C adjacency.

Bit #	Adjacency	Owner	Example Bit Setting
1	I->A	I	1
2	A->B	A	1
	B->A	B	
3	I->C	I	0
4	A->C	A	1
5	B->D	B	1
6	C->D	C	1
	D->C	D	
7	C->E	C	1
8	D->E	D	0

Replication and Elimination Protecting A-&gt;C

Table 1: Controlling Replication

- o The BIER header with the controlling BitString is injected in the packet by the ingress node of the deterministic path. That node may act as a replication point, in which case it may issue multiple copies of the packet

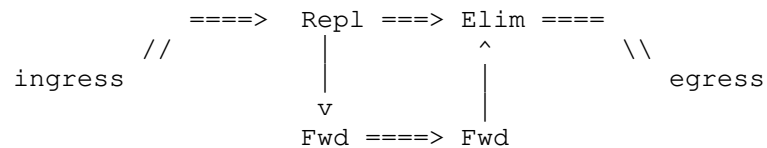


Figure 11: Enabled Adjacencies

- o For each of its bits that is set in the BIER header, the owner replication point resets the bit and transmits towards the associated adjacency; to achieve this, the replication point copies the packet and inserts the relevant data plane information, such as a source route header, towards the adjacency that corresponds to the bit

Adjacency	BIER BitString
I->A	01011110
A->B	00011110
B->D	00010110
D->C	00010010
A->C	01001110

BitString in BIER Header as Packet Progresses

Table 2: BIER-TE in Action

- o Adversely, an elimination node on the way strips the data plane information and performs a bitwise AND on the BitStrings from the various copies of the packet that it has received, before it forwards the packet with the resulting BitString.

Operation	BIER BitString
D->C	00010010
A->C	01001110
AND in C	00000010
C->E	00000000

BitString Processing at Elimination Point C

Table 3: BIER-TE in Action (cont.)

- o In this example, all the transmissions succeeded and the BitString at arrival has all the bits reset - note that the egress may be an Elimination Point in which case this is evaluated after this node has performed its AND operation on the received BitStrings).

Failing Adjacency	Egress BIER BitString
I->A	Frame Lost
I->B	Not Tried
A->C	00010000
A->B	01001100
B->D	01001100
D->C	01001100
C->E	Frame Lost
D->E	Not Tried

BitString indicating failures

Table 4: BIER-TE in Action (cont.)

- o But if a transmission failed along the way, one (or more) bit is never cleared. Table 4 provides the possible outcomes of a transmission. If the frame is lost, then it is probably due to a failure in either I->A or C->E, and the controller should enable I->B and D->E to find out. A BitString of 00010000 indicates unequivocally a transmission error on the A->C adjacency, and a BitString of 01001100 indicates a loss in either A->B, B->D or D->C; enabling D->E on the next packets may provide more information to sort things out.

In more details:

The BIER header is of variable size, and a DetNet network of a limited size can use a model with 64 bits if 64 adjacencies are enough, whereas a larger deployment may be able to signal up to 256 adjacencies for use in very complex paths. Figure 7 illustrates a BIER header as encapsulated within MPLS. The format of this header is common to BIER and BIER-TE.

For the DetNet data plane, a replication point is an ingress point for more than one adjacency, and an elimination point is an egress point for more than one adjacency.

A pre-populated state in a replication node indicates which bits are served by this node and to which adjacency each of these bits corresponds. With DetNet, the state is typically installed by a controller entity such as a PCE. The way the adjacency is signaled in the packet is fully abstracted in the bit representation and must be provisioned to the replication nodes and maintained as a local state, together with the timing or shaping information for the associated flow.



The DetNet data plane uses BIER-TE to control which adjacencies are used for a given packet. This is signaled from the path ingress, which sets the appropriate bits in the BIER BitString to indicate which replication must happen.

The replication point clears the bit associated to the adjacency where the replica is placed, and the elimination points perform a logical AND of the BitStrings of the copies that it gets before forwarding.

As is apparent in the examples above, clearing the bits enables to trace a packet to the replication points that made any particular copy. BIER-TE also enables to detect the failing adjacencies or sequences of adjacencies along a path and to activate additional replications to counter balance the failures.

Finally, using the same BIER-TE bit for both directions of the steps of the ladder enables to avoid replication in both directions along the crossing adjacencies. At the time of sending along the step of the ladder, the bit may have been already reset by performing the AND operation with the copy from the other side, in which case the transmission is not needed and does not occur (since the control bit is now off).

#### 4.2.8. Higher layer header fields

Fields of headers belonging to higher OSI layers can be used to implement functionality that is not provided e.g., by the IPv6 or IPv4 header fields. However, this approach cannot be always applied, e.g., due to encryption. Furthermore, even if this approach is applicable, it requires deep packet inspection from the routers and switches. There are implementation dependent limits how far into the packet the lookup can be done efficiently in the fast path. In general a safe bet is between 128 and 256 octets for the maximum lookup depth. Various higher layer protocols can be applied. Some examples are provided here for the sequence numbering feature (Section 3.4).

##### 4.2.8.1. TCP

The TCP header includes a sequence number parameter, which can be applied to detect and eliminate duplicate packets if seamless redundancy is used. As the TCP header is right after the IP header, it does not require very deep packet inspection; the 4-byte sequence number is conveyed by bits 32 through 63 of the TCP header. In addition to sequencing, the TCP header also contain source and destination port information that can be used for assisting the flow identification.

#### 4.2.8.2. RTP

RTP is often used to deliver time critical traffic in IP networks. RTP is carried on top of IP and UDP [RFC3550]. The RTP header includes a 2-byte sequence number, which can be used to detect and eliminate duplicate packets if seamless redundancy is used. The sequence number is conveyed by bits 16 through 31 of the RTP header. In addition to the sequence number the RTP header has also timestamp field (bits 32 through 63) that can be useful for time synchronization purposes. Furthermore, the RTP header has also one or more synchronization sources (bits starting from 64) that can potentially be useful for flow identification purposes.

#### 5. Summary of data plane alternatives

TBD.

#### 6. Security considerations

TBD.

#### 7. IANA Considerations

This document has no IANA considerations.

#### 8. Acknowledgements

The author(s) ACK and NACK.

The following people were part of the DetNet Data Plane Design Team:

Jouni Korhonen  
Janos Farkas  
Norman Finn  
Olivier Marce  
Gregory Mirsky  
Pascal Thubert  
Zhuangyan Zhuang

The DetNet chairs serving during the DetNet Data Plane Design Team:

Lou Berger  
Pat Thaler

## 9. References

### 9.1. Informative References

- [I-D.eckert-bier-te-arch]  
Eckert, T. and G. Cauchie, "Traffic Engineering for Bit Index Explicit Replication BIER-TE", draft-eckert-bier-te-arch-02 (work in progress), October 2015.
- [I-D.finn-detnet-architecture]  
Finn, N., Thubert, P., and M. Teener, "Deterministic Networking Architecture", draft-finn-detnet-architecture-02 (work in progress), November 2015.
- [I-D.finn-detnet-problem-statement]  
Finn, N. and P. Thubert, "Deterministic Networking Problem Statement", draft-finn-detnet-problem-statement-04 (work in progress), October 2015.
- [I-D.ietf-6man-rfc2460bis]  
Deering, S. and B. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", draft-ietf-6man-rfc2460bis-04 (work in progress), March 2016.
- [I-D.ietf-6man-segment-routing-header]  
Previdi, S., Filsfils, C., Field, B., Leung, I., Linkova, J., Kosugi, T., Vyncke, E., and D. Lebrun, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-01 (work in progress), March 2016.
- [I-D.ietf-bier-architecture]  
Wijnands, I., Rosen, E., Dolganow, A., P, T., and S. Aldrin, "Multicast using Bit Index Explicit Replication", draft-ietf-bier-architecture-03 (work in progress), January 2016.
- [I-D.ietf-idr-ls-distribution]  
Gredler, H., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and TE Information using BGP", draft-ietf-idr-ls-distribution-13 (work in progress), October 2015.
- [I-D.ietf-intarea-gre-ipv6]  
Pignataro, C., Bonica, R., and S. Krishnan, "IPv6 Support for Generic Routing Encapsulation (GRE)", draft-ietf-intarea-gre-ipv6-14 (work in progress), September 2015.

- [I-D.ietf-isis-pcr]  
Farkas, J., Bragg, N., Unbehagen, P., Parsons, G.,  
Ashwood-Smith, P., and C. Bowers, "IS-IS Path Computation  
and Reservation", draft-ietf-isis-pcr-05 (work in  
progress), February 2016.
- [I-D.ietf-mpsls-residence-time]  
Mirsky, G., Ruffini, S., Gray, E., Drake, J., Bryant, S.,  
and S. Sasha, "Residence Time Measurement in MPLS  
network", draft-ietf-mpsls-residence-time-05 (work in  
progress), March 2016.
- [I-D.ietf-spring-segment-routing]  
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S.,  
and R. Shakir, "Segment Routing Architecture", draft-ietf-  
spring-segment-routing-07 (work in progress), December  
2015.
- [I-D.ietf-sunset4-gapanalysis]  
Perreault, S., Tsou, T., Zhou, C., and P. Fan, "Gap  
Analysis for IPv4 Sunset", draft-ietf-  
sunset4-gapanalysis-07 (work in progress), April 2015.
- [I-D.ietf-v6ops-ipv6-ehs-in-real-world]  
Gont, F., Linkova, J., Chown, T., and S. LIU,  
"Observations on the Dropping of Packets with IPv6  
Extension Headers in the Real World", draft-ietf-v6ops-  
ipv6-ehs-in-real-world-02 (work in progress), December  
2015.
- [IEEE802.1Qbv]  
IEEE, "Enhancements for Scheduled Traffic", 2016,  
<<http://www.ieee802.org/1/files/private/bv-drafts/>>.
- [IEEE802.1Qch]  
IEEE, "Cyclic Queuing and Forwarding", 2016,  
<<http://www.ieee802.org/1/files/private/ch-drafts/>>.
- [IEEE8021CB]  
Finn, N., "Draft Standard for Local and metropolitan area  
networks - Seamless Redundancy", IEEE P802.1CB  
/D2.1 P802.1CB, December 2015,  
<[http://www.ieee802.org/1/files/private/cb-drafts/  
d2/802-1CB-d2-1.pdf](http://www.ieee802.org/1/files/private/cb-drafts/d2/802-1CB-d2-1.pdf)>.

- [IEEE8021Qca] IEEE 802.1, "IEEE 802.1Qca Bridges and Bridged Networks - Amendment 24: Path Control and Reservation", IEEE P802.1Qca/D2.1 P802.1Qca, June 2015, <<http://www.ieee802.org/1/files/private/ca-drafts/d2/802-1Qca-d2-1.pdf>>.
- [RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<http://www.rfc-editor.org/info/rfc791>>.
- [RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, DOI 10.17487/RFC1122, October 1989, <<http://www.rfc-editor.org/info/rfc1122>>.
- [RFC1393] Malkin, G., "Traceroute Using an IP Option", RFC 1393, DOI 10.17487/RFC1393, January 1993, <<http://www.rfc-editor.org/info/rfc1393>>.
- [RFC1700] Reynolds, J. and J. Postel, "Assigned Numbers", RFC 1700, DOI 10.17487/RFC1700, October 1994, <<http://www.rfc-editor.org/info/rfc1700>>.
- [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, <<http://www.rfc-editor.org/info/rfc2205>>.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <<http://www.rfc-editor.org/info/rfc2460>>.
- [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, DOI 10.17487/RFC2474, December 1998, <<http://www.rfc-editor.org/info/rfc2474>>.
- [RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, DOI 10.17487/RFC2702, September 1999, <<http://www.rfc-editor.org/info/rfc2702>>.
- [RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, DOI 10.17487/RFC2784, March 2000, <<http://www.rfc-editor.org/info/rfc2784>>.

- [RFC2890] Dommety, G., "Key and Sequence Number Extensions to GRE", RFC 2890, DOI 10.17487/RFC2890, September 2000, <<http://www.rfc-editor.org/info/rfc2890>>.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<http://www.rfc-editor.org/info/rfc3031>>.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <<http://www.rfc-editor.org/info/rfc3032>>.
- [RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <<http://www.rfc-editor.org/info/rfc3168>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<http://www.rfc-editor.org/info/rfc3209>>.
- [RFC3270] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, DOI 10.17487/RFC3270, May 2002, <<http://www.rfc-editor.org/info/rfc3270>>.
- [RFC3443] Agarwal, P. and B. Akyol, "Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks", RFC 3443, DOI 10.17487/RFC3443, January 2003, <<http://www.rfc-editor.org/info/rfc3443>>.
- [RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <<http://www.rfc-editor.org/info/rfc3473>>.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <<http://www.rfc-editor.org/info/rfc3550>>.

- [RFC3931] Lau, J., Ed., Townsley, M., Ed., and I. Goyret, Ed., "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, DOI 10.17487/RFC3931, March 2005, <<http://www.rfc-editor.org/info/rfc3931>>.
- [RFC3985] Bryant, S., Ed. and P. Pate, Ed., "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <<http://www.rfc-editor.org/info/rfc3985>>.
- [RFC4023] Worster, T., Rekhter, Y., and E. Rosen, Ed., "Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)", RFC 4023, DOI 10.17487/RFC4023, March 2005, <<http://www.rfc-editor.org/info/rfc4023>>.
- [RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, DOI 10.17487/RFC4385, February 2006, <<http://www.rfc-editor.org/info/rfc4385>>.
- [RFC4446] Martini, L., "IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)", BCP 116, RFC 4446, DOI 10.17487/RFC4446, April 2006, <<http://www.rfc-editor.org/info/rfc4446>>.
- [RFC4447] Martini, L., Ed., Rosen, E., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", RFC 4447, DOI 10.17487/RFC4447, April 2006, <<http://www.rfc-editor.org/info/rfc4447>>.
- [RFC4448] Martini, L., Ed., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, DOI 10.17487/RFC4448, April 2006, <<http://www.rfc-editor.org/info/rfc4448>>.
- [RFC4664] Andersson, L., Ed. and E. Rosen, Ed., "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, DOI 10.17487/RFC4664, September 2006, <<http://www.rfc-editor.org/info/rfc4664>>.
- [RFC4817] Townsley, M., Pignataro, C., Wainner, S., Seely, T., and J. Young, "Encapsulation of MPLS over Layer 2 Tunneling Protocol Version 3", RFC 4817, DOI 10.17487/RFC4817, March 2007, <<http://www.rfc-editor.org/info/rfc4817>>.

- [RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <<http://www.rfc-editor.org/info/rfc4875>>.
- [RFC5086] Vainshtein, A., Ed., Sasson, I., Metz, E., Frost, T., and P. Pate, "Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)", RFC 5086, DOI 10.17487/RFC5086, December 2007, <<http://www.rfc-editor.org/info/rfc5086>>.
- [RFC5087] Stein, Y(J)., Shashoua, R., Insler, R., and M. Anavi, "Time Division Multiplexing over IP (TDMoIP)", RFC 5087, DOI 10.17487/RFC5087, December 2007, <<http://www.rfc-editor.org/info/rfc5087>>.
- [RFC5129] Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, DOI 10.17487/RFC5129, January 2008, <<http://www.rfc-editor.org/info/rfc5129>>.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <<http://www.rfc-editor.org/info/rfc5305>>.
- [RFC5331] Aggarwal, R., Rekhter, Y., and E. Rosen, "MPLS Upstream Label Assignment and Context-Specific Label Space", RFC 5331, DOI 10.17487/RFC5331, August 2008, <<http://www.rfc-editor.org/info/rfc5331>>.
- [RFC5332] Eckert, T., Rosen, E., Ed., Aggarwal, R., and Y. Rekhter, "MPLS Multicast Encapsulations", RFC 5332, DOI 10.17487/RFC5332, August 2008, <<http://www.rfc-editor.org/info/rfc5332>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<http://www.rfc-editor.org/info/rfc5440>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<http://www.rfc-editor.org/info/rfc5462>>.



- [RFC5586] Bocci, M., Ed., Vigoureux, M., Ed., and S. Bryant, Ed., "MPLS Generic Associated Channel", RFC 5586, DOI 10.17487/RFC5586, June 2009, <<http://www.rfc-editor.org/info/rfc5586>>.
- [RFC5659] Bocci, M. and S. Bryant, "An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge", RFC 5659, DOI 10.17487/RFC5659, October 2009, <<http://www.rfc-editor.org/info/rfc5659>>.
- [RFC5921] Bocci, M., Ed., Bryant, S., Ed., Frost, D., Ed., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", RFC 5921, DOI 10.17487/RFC5921, July 2010, <<http://www.rfc-editor.org/info/rfc5921>>.
- [RFC5960] Frost, D., Ed., Bryant, S., Ed., and M. Bocci, Ed., "MPLS Transport Profile Data Plane Architecture", RFC 5960, DOI 10.17487/RFC5960, August 2010, <<http://www.rfc-editor.org/info/rfc5960>>.
- [RFC6073] Martini, L., Metz, C., Nadeau, T., Bocci, M., and M. Aissaoui, "Segmented Pseudowire", RFC 6073, DOI 10.17487/RFC6073, January 2011, <<http://www.rfc-editor.org/info/rfc6073>>.
- [RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <<http://www.rfc-editor.org/info/rfc6275>>.
- [RFC6371] Busi, I., Ed. and D. Allan, Ed., "Operations, Administration, and Maintenance Framework for MPLS-Based Transport Networks", RFC 6371, DOI 10.17487/RFC6371, September 2011, <<http://www.rfc-editor.org/info/rfc6371>>.
- [RFC6373] Andersson, L., Ed., Berger, L., Ed., Fang, L., Ed., Bitar, N., Ed., and E. Gray, Ed., "MPLS Transport Profile (MPLS-TP) Control Plane Framework", RFC 6373, DOI 10.17487/RFC6373, September 2011, <<http://www.rfc-editor.org/info/rfc6373>>.
- [RFC6378] Weingarten, Y., Ed., Bryant, S., Osborne, E., Sprecher, N., and A. Fulignoli, Ed., "MPLS Transport Profile (MPLS-TP) Linear Protection", RFC 6378, DOI 10.17487/RFC6378, October 2011, <<http://www.rfc-editor.org/info/rfc6378>>.

- [RFC6426] Gray, E., Bahadur, N., Boutros, S., and R. Aggarwal, "MPLS On-Demand Connectivity Verification and Route Tracing", RFC 6426, DOI 10.17487/RFC6426, November 2011, <<http://www.rfc-editor.org/info/rfc6426>>.
- [RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <<http://www.rfc-editor.org/info/rfc6437>>.
- [RFC6540] George, W., Donley, C., Liljenstolpe, C., and L. Howard, "IPv6 Support Required for All IP-Capable Nodes", BCP 177, RFC 6540, DOI 10.17487/RFC6540, April 2012, <<http://www.rfc-editor.org/info/rfc6540>>.
- [RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", RFC 6564, DOI 10.17487/RFC6564, April 2012, <<http://www.rfc-editor.org/info/rfc6564>>.
- [RFC6621] Macker, J., Ed., "Simplified Multicast Forwarding", RFC 6621, DOI 10.17487/RFC6621, May 2012, <<http://www.rfc-editor.org/info/rfc6621>>.
- [RFC6658] Bryant, S., Ed., Martini, L., Swallow, G., and A. Malis, "Packet Pseudowire Encapsulation over an MPLS PSN", RFC 6658, DOI 10.17487/RFC6658, July 2012, <<http://www.rfc-editor.org/info/rfc6658>>.
- [RFC6718] Muley, P., Aissaoui, M., and M. Bocci, "Pseudowire Redundancy", RFC 6718, DOI 10.17487/RFC6718, August 2012, <<http://www.rfc-editor.org/info/rfc6718>>.
- [RFC6733] Fajardo, V., Ed., Arkko, J., Loughney, J., and G. Zorn, Ed., "Diameter Base Protocol", RFC 6733, DOI 10.17487/RFC6733, October 2012, <<http://www.rfc-editor.org/info/rfc6733>>.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, DOI 10.17487/RFC6790, November 2012, <<http://www.rfc-editor.org/info/rfc6790>>.
- [RFC6814] Pignataro, C. and F. Gont, "Formally Deprecating Some IPv4 Options", RFC 6814, DOI 10.17487/RFC6814, November 2012, <<http://www.rfc-editor.org/info/rfc6814>>.

- [RFC6864] Touch, J., "Updated Specification of the IPv4 ID Field", RFC 6864, DOI 10.17487/RFC6864, February 2013, <<http://www.rfc-editor.org/info/rfc6864>>.
- [RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, DOI 10.17487/RFC7045, December 2013, <<http://www.rfc-editor.org/info/rfc7045>>.
- [RFC7167] Frost, D., Bryant, S., Bocci, M., and L. Berger, "A Framework for Point-to-Multipoint MPLS in Transport Networks", RFC 7167, DOI 10.17487/RFC7167, April 2014, <<http://www.rfc-editor.org/info/rfc7167>>.
- [RFC7209] Sajassi, A., Aggarwal, R., Uttaro, J., Bitar, N., Henderickx, W., and A. Isaac, "Requirements for Ethernet VPN (EVPN)", RFC 7209, DOI 10.17487/RFC7209, May 2014, <<http://www.rfc-editor.org/info/rfc7209>>.
- [RFC7271] Ryoo, J., Ed., Gray, E., Ed., van Helvoort, H., D'Alessandro, A., Cheung, T., and E. Osborne, "MPLS Transport Profile (MPLS-TP) Linear Protection to Match the Operational Expectations of Synchronous Digital Hierarchy, Optical Transport Network, and Ethernet Transport Network Operators", RFC 7271, DOI 10.17487/RFC7271, June 2014, <<http://www.rfc-editor.org/info/rfc7271>>.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <<http://www.rfc-editor.org/info/rfc7348>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<http://www.rfc-editor.org/info/rfc7399>>.
- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", RFC 7426, DOI 10.17487/RFC7426, January 2015, <<http://www.rfc-editor.org/info/rfc7426>>.

- [RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <<http://www.rfc-editor.org/info/rfc7432>>.
- [RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, DOI 10.17487/RFC7510, April 2015, <<http://www.rfc-editor.org/info/rfc7510>>.
- [RFC7637] Garg, P., Ed. and Y. Wang, Ed., "NVGRE: Network Virtualization Using Generic Routing Encapsulation", RFC 7637, DOI 10.17487/RFC7637, September 2015, <<http://www.rfc-editor.org/info/rfc7637>>.
- [ST20227] SMPTE 2022, "Seamless Protection Switching of SMPTE ST 2022 IP Datagrams", ST 2022-7:2013, 2013, <<https://www.smpte.org/digital-library>>.
- [TSNTG] IEEE Standards Association, "IEEE802.1 Time-Sensitive Networks Task Group", 2013, <<http://www.IEEE802.org/1/pages/avbridges.html>>.

## 9.2. URIs

- [1] <http://6lab.cisco.com/stats/>
- [2] <https://www.google.com/intl/en/ipv6/statistics.html>
- [3] <https://datatracker.ietf.org/wg/spring/charter/>
- [4] <http://www.iana.org/assignments/g-ach-parameters/g-ach-parameters.xhtml>
- [5] <http://ftp.isi.edu/in-notes/iana/assignments/ethernet-numbers>
- [6] <https://tools.ietf.org/wg/bess/>

Appendix A. Examples of combined DetNet Service and Transport layers

Authors' Addresses

Jouni Korhonen (editor)  
Broadcom  
3151 Zanker Road  
San Jose, CA 95134  
USA

Email: jouni.nospam@gmail.com

Janos Farkas  
Ericsson  
Konyves Kalman krt. 11/B  
Budapest 1097  
Hungary

Email: janos.farkas@ericsson.com

Gregory Mirsky  
Ericsson

Email: gregory.mirsky@ericsson.com

Pascal Thubert  
Cisco

Email: pthubert@cisco.com

Yan Zhuang  
Huawei

Email: zhuangyan.zhuang@huawei.com

Lou Berger  
LabN Consulting, L.L.C.

Email: lberger@labn.net

DetNet  
Internet-Draft  
Intended status: Standards Track  
Expires: September 4, 2016

N. Finn  
P. Thubert  
Cisco  
M. Johas Teener  
Broadcom  
March 3, 2016

Deterministic Networking Architecture  
draft-finn-detnet-architecture-03

Abstract

Deterministic Networking (DetNet) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low data loss rates and bounded latency. Techniques used include: 1) reserving data plane resources for individual (or aggregated) DetNet flows in some or all of the relay systems (bridges or routers) along the path of the flow; 2) providing fixed paths for DetNet flows that do not rapidly change with the network topology; and 3) sequentializing, replicating, and eliminating duplicate packets at various points to ensure the availability of at least one path. The capabilities can be managed by configuration, or by manual or automatic network management.

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 <http://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 September 4, 2016.

Copyright Notice

Copyright (c) 2016 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

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Providing the DetNet Quality of Service . . . . .	5
3.1. Zero Congestion Loss . . . . .	6
3.2. Pinned paths . . . . .	7
3.3. Seamless Redundancy . . . . .	7
4. DetNet Architecture . . . . .	9
4.1. Elements of DetNet Architecture . . . . .	9
4.2. Traffic Engineering for DetNet . . . . .	10
4.2.1. The Application Plane . . . . .	13
4.2.2. The Controller Plane . . . . .	13
4.2.3. The Network Plane . . . . .	14
4.3. DetNet flows . . . . .	15
4.3.1. Source guarantees . . . . .	15
4.3.2. Incomplete Networks . . . . .	16
4.4. Queuing, Shaping, Scheduling, and Preemption . . . . .	16
4.5. Coexistence with normal traffic . . . . .	17
4.6. Fault Mitigation . . . . .	18
4.7. Protocol Stack Model . . . . .	18
4.8. Advertising resources, capabilities and adjacencies . . . . .	20
4.9. Provisioning model . . . . .	20
4.9.1. Centralized Path Computation and Installation . . . . .	20
4.9.2. Distributed Path Setup . . . . .	21
4.10. Scaling to larger networks . . . . .	21
4.11. Connected islands vs. networks . . . . .	21
5. Compatibility with Layer-2 . . . . .	22
6. Open Questions . . . . .	22
6.1. Data plane shapers and schedulers . . . . .	22
6.2. DetNet flow identification and sequencing . . . . .	22
6.3. Flat vs. hierarchical control . . . . .	23
6.4. Peer-to-peer reservation protocol . . . . .	23
7. Security Considerations . . . . .	24
8. IANA Considerations . . . . .	24
9. Acknowledgements . . . . .	24
10. Access to IEEE 802.1 documents . . . . .	24
11. Informative References . . . . .	25

Authors' Addresses	28
--------------------	----

## 1. Introduction

Deterministic Networking (DetNet) is a service that can be offered by a network to data flows (DetNet flows) that are limited, at their source, to a maximum data rate specified by that source. DetNet provides these flows extremely low packet loss rates and assured maximum end-to-end delivery latency. This is accomplished by dedicating network resources such as link bandwidth and buffer space to DetNet flows and/or classes of DetNet flows. Unused reserved resources are available to non-DetNet packets.

The Deterministic Networking Problem Statement

[I-D.finn-detnet-problem-statement] introduces Deterministic Networking, and Deterministic Networking Use Cases

[I-D.grossman-detnet-use-cases] summarizes the need for it.

A goal of DetNet is a converged network in all respects. That is, the presence of DetNet flows does not preclude non-DetNet flows, and the benefits offered DetNet flows should not, except in extreme cases, prevent existing QoS mechanisms from operating in a normal fashion, subject to the bandwidth required for the DetNet flows. A single source-destination pair can trade both DetNet and non-DetNet flows. End systems and applications need not instantiate special interfaces for DetNet flows. Networks are not restricted to certain topologies; connectivity is not restricted. Any application that generates a data flow that can be usefully characterized as having a maximum bandwidth should be able to take advantage of DetNet, as long as the necessary resources can be reserved. Reservations can be made by the application itself, via network management, by an applications controller, or by other means.

Many applications of interest to Deterministic Networking require the ability to synchronize the clocks in end systems to a sub-microsecond accuracy. Some of the queue control techniques defined in Section 4.4 also require time synchronization among relay systems. The means used to achieve time synchronization are not addressed in this document.

The present document is an individual contribution, intended by the authors for eventual adoption by the DetNet working group. As such, it expresses the only the opinions of the authors.



## 2. Terminology

The following special terms are used in this document in order to avoid the assumption that a given element in the architecture does or does not have Internet Protocol stack, functions as a router, bridge, firewall, or otherwise plays a particular role at Layer-2 or higher. This section also serves as a dictionary for translating between IEEE 802 and DetNet terminology.

### destination

An end system capable of sinking a DetNet flow.

### DetNet flow

A DetNet flow is a sequence of packets from a single source, through some number of relay systems to one or more destinations, that is limited by the source in its maximum packet size and transmission rate, and can thus be ensured the DetNet Quality of Service (QoS) from the network.

### end system

Commonly called a "host" in IETF documents, and an "end station" in IEEE 802 documents. End systems of interest to this document are either sources or destinations.

### listener

The IEEE 802 term for a destination of a DetNet flow.

### relay system

A router, bridge, Label Switch Router (LSR), firewall, or any other system that forwards packets from one interface to another.

### reservation

A trail of configuration from source to destination(s) through relay systems associated with a DetNet flow, required to deliver the benefits of DetNet.

### source

An end system capable of sourcing a DetNet flow.

### stream

The IEEE 802 term for a DetNet flow.

### talker

The IEEE 802 term for the source of a DetNet flow.

### 3. Providing the DetNet Quality of Service

DetNet Quality of Service is expressed in terms of:

- o Minimum and maximum end-to-end latency from source to destination;
- o Probability of loss of a packet, assuming the normal operation of the relay systems and links;
- o Probability of loss of a packet in the event of the failure of a relay system or link.

It is a distinction of DetNet that it is concerned solely with worst-case values for all of the above parameters. Average, mean, or typical values are of no interest, because they do not affect the ability of a real-time system to perform its tasks. For example, in this document, we will often speak of assuring a DetNet flow a bounded latency. In general, a trivial priority-based queuing scheme will give better average latency to a data flow than DetNet, but of course, the worst-case latency is essentially unbounded.

Three techniques are employed by DetNet to achieve these QoS parameters:

- a. Zero congestion loss (Section 3.1). Network resources such as link bandwidth, buffers, queues, shapers, and scheduled input/output slots are assigned in each relay system to the use of a specific DetNet flow or class of DetNet flows. Given a finite amount of buffer space, zero congestion loss necessarily ensures a bounded end-to-end latency. Depending on the resources employed, a minimum latency, and thus bounded jitter, can also be achieved.
- b. Pinned paths (Section 3.2). Point-to-point paths or point-to-multipoint trees through the network from a source to one or more destinations can be established, and DetNet flows assigned to follow a particular path or tree.
- c. Packet replication and deletion (Section 3.3). End systems and/or relay systems can number packets sequentially, replicate them, and later eliminate all but one of the replicants, at multiple points in the network in order to ensure that one (or more) equipment failure events still leave at least one path intact for a DetNet flow.

These three techniques can be applied independently, giving eight possible combinations, including none (no DetNet), although some combinations are of wider utility than others. This separation keeps

the protocol stack coherent and maximizes interoperability with existing and developing standards in this (IETF) and other Standards Development Organizations. Some examples of typical expected combinations:

- o Pinned paths (a) plus packet replication (b) are exactly the techniques employed by [HSR-PRP]. Pinned paths are achieved by limiting the physical topology of the network, and the sequentialization, replication, and duplicate elimination are facilitated by packet tags added at the front or the end of Ethernet frames.
- o Zero congestion loss (a) alone is offered by IEEE 802.1 Audio Video bridging [IEEE802.1BA-2011]. As long as the network suffers no failures, zero congestion loss can be achieved through the use of a reservation protocol (MSRP), shapers in every relay system (bridge), and a bit of network calculus.
- o Using all three together gives maximum protection.

There are, of course, simpler methods available (and employed, today) to achieve levels of latency and packet loss that are satisfactory for many applications. Prioritization and over-provisioning is one such technique. However, these methods generally work best in the absence of any significant amount of non-critical traffic in the network (if, indeed, such traffic is supported at all), or work only if the critical traffic constitutes only a small portion of the network's theoretical capacity, or work only if all systems are functioning properly, or in the absence of actions by end systems that disrupt the network's operations.

There are any number of methods in use, defined, or in progress for accomplishing each of the above techniques. It is expected that this DetNet Architecture will assist various vendors, users, and/or "vertical" Standards Development Organizations (dedicated to a single industry) to make selections among the available means of implementing DetNet networks.

### 3.1. Zero Congestion Loss

The primary means by which DetNet achieves its QoS assurances is to completely eliminate congestion at an output port as a cause of packet loss. Given that a DetNet flow cannot be throttled, this can be achieved only by the provision of sufficient buffer storage at each hop through the network to ensure that no packets are dropped due to a lack of buffer storage.

Ensuring adequate buffering requires, in turn, that the source, and every relay system along the path to the destination (or nearly every relay system -- see Section 4.3.2) be careful to regulate its output to not exceed the data rate for any DetNet flow, except for brief periods when making up for interfering traffic. Any packet sent ahead of its time potentially adds to the number of buffers required by the next hop, and may thus exceed the resources allocated for a particular DetNet flow.

The low-level mechanisms described in Section 4.4 provide the necessary regulation of transmissions by an edge system or relay system to ensure zero congestion loss. The reservation of the bandwidth and buffers for a DetNet flow requires the provisioning described in Section 4.9.

### 3.2. Pinned paths

In networks controlled by typical peer-to-peer protocols such as IEEE 802.1 ISIS bridged networks or IETF OSPF routed networks, a network topology event in one part of the network can impact, at least briefly, the delivery of data in parts of the network remote from the failure or recovery event. Thus, even redundant paths through a network, if controlled by the typical peer-to-peer protocols, do not eliminate the chances of brief losses of contact.

Many real-time networks rely on physical rings or chains of two-port devices, with a relatively simple ring control protocol. This supports redundant paths with a minimum of wiring. As an additional benefit, ring topologies can often utilize different topology management protocols than those used for a mesh network, with a consequent reduction in the response time to topology changes. Of course, this comes at some cost in terms of increased hop count, and thus latency, for the typical path.

In order to get the advantages of low hop count and still ensure against even very brief losses of connectivity, DetNet employs pinned paths, where the path taken by a given DetNet flow does not change, at least immediately, and likely not at all, in response to network topology events. When combined with seamless redundancy (Section 3.3), this results in a high likelihood of continuous connectivity.

### 3.3. Seamless Redundancy

After congestion loss has been eliminated, the most important causes of packet loss are random media and/or memory faults, and equipment failures.

Seamless redundancy involves three capabilities:

- o Adding sequence numbers, once, to the packets of a DetNet flow.
- o Replicating these packets and, typically, sending them along at least two different paths to the destination(s). (Often, the pinned paths of Section 3.2.)
- o Discarding duplicated packets.

In the simplest case, this amounts to replicating each packet in a source that has two interfaces, and conveying them through the network, along separate paths, to the similarly dual-homed destinations, that discard the extras. This ensures that one path (with zero congestion loss) remains, even if some relay system fails.

Alternatively, relay systems in the network can provide replication and elimination facilities at various points in the network, so that multiple failures can be accommodated.

This is shown in the following figure, where the two relay systems each replicate (R) the DetNet flow on input, sending the DetNet flow to both the other relay system and to the end system, and eliminate duplicates (E) on the output interface to the right-hand end system. Any one link in the network can fail, and the DetNet flow can still get through. Furthermore, two links can fail, as long as they are in different segments of the network.

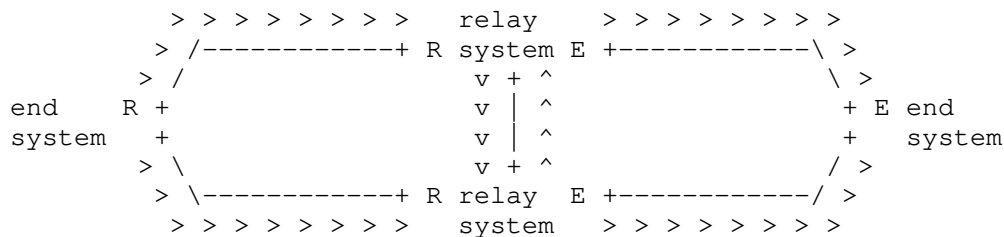


Figure 1

Note that seamless redundancy does not react to and correct failures; it is entirely passive. Thus, intermittent failures, mistakenly created access control lists, or misrouted data is handled just the same as the equipment failures that are detected handled by typical routing and bridging protocols.

## 4. DetNet Architecture

### 4.1. Elements of DetNet Architecture

The DetNet architecture has a number of elements, discussed in the following sections. Note that not every application requires all of these elements.

- a. A model for the definition, identification, and operation of DetNet flows (Section 4.3), for use by relay systems to classify and process individual packets following per-flow rules.
- b. A model for the flow of data out of an end system or through a relay system that can be used to predict the bounds for that system's impact on the QoS of a DetNet flow, for use by the Controllers to configure policing and shaping engines in Network Systems over the Southbound interface. The model includes:
  1. A model for queuing, transmission selection, shaping, preemption, and timing resources that can be used by an end system or relay system to control the selection of packets output on an interface. These models must have sufficiently well-defined characteristics, both individually and in the aggregate, to give predictable results for the QoS for DetNet packets (Section 4.4).
  2. A model for identifying misbehaving DetNet flows and mitigating their impact on properly functioning DetNet flows (Section 4.6).
- c. A model for the relay system to inform the controller(s) of the information it needs for adequate path computations (Section 4.2) including:
  1. Systems' individual capabilities (e.g. can do replication, can do precise time).
  2. Link capabilities and resources (e.g. bandwidth, transmission delay, hardware deterministic support to the physical layer, ...)
  3. Physical resources (total and available buffers, timers, queues, etc)
  4. Network Adjacencies (neighbors)

- d. A model for the provision of a service, by end systems or relay systems, to replicate and forward a DetNet flow over redundant paths. The model includes:
  - 1. A model for specifying multiple stable paths across a network that can perform packet forwarding at both Layer 3 and at lower layers, to which specific DetNet flows can be assigned (Section 4.2).
  - 2. A model and data plane format(s) for sequencing and replicating the packets of a DetNet flow, typically at or near the source, sending the replicated DetNet flows over different stable paths, merging and/or re-replicating those packets at other points in the network, and finally eliminating the duplicates, typically at or near the destination(s), in order to provide high availability (Section 3.3).
- e. The protocol stack model for an end system and/or a relay system should support the above elements in a manner that maximizes the applicability of existing standards and protocols to the DetNet problem, and allows for the creation of new protocols only where needed, thus making DetNet an add-on feature to existing networks, rather than a new way to do networking. In particular this protocol stack supports networks in which the path from source to destination(s) includes bridges and/or routers in any order (Section 4.7).
- f. A variety of models for the provisioning of DetNet flows can be envisioned, including orchestration by a central controller or by a federation of controllers, provisioning by relay systems and end systems sharing peer-to-peer protocols, by off-line configuration, or by a combination of these methods. The provisioning models are similar to existing Layer-2 and Layer-3 models, in order to minimize the amount of innovation required in this area (Section 4.9).

#### 4.2. Traffic Engineering for DetNet

Traffic Engineering Architecture and Signaling (TEAS) [TEAS] defines traffic-engineering architectures for generic applicability across packet and non-packet networks. From TEAS perspective, Traffic Engineering (TE) refers to techniques that enable operators to control how specific traffic flows are treated within their networks.

Because of its very nature of establishing pinned optimized paths, Deterministic Networking can be seen as a new, specialized branch of

Traffic Engineering, and inherits its architecture with a separation into planes.

The Deterministic Networking architecture is thus composed of three planes, a (User) Application Plane, a Controller Plane, and a Network Plane, which echoes that of Software-Defined Networking (SDN): Layers and Architecture Terminology [RFC7426] which is represented below:



## SDN Layers and Architecture Terminology per RFC 7426

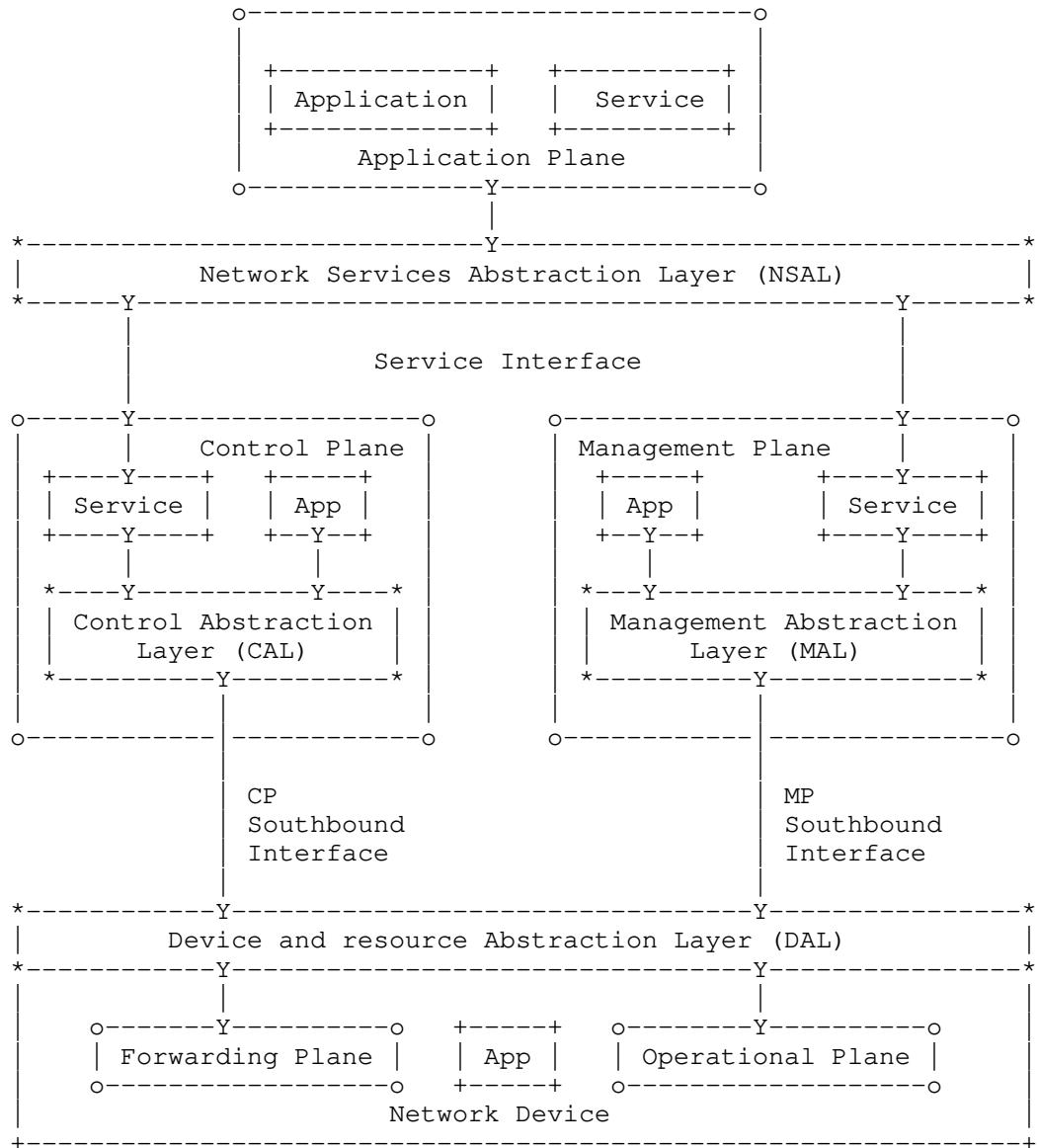


Figure 2

#### 4.2.1. The Application Plane

Per [RFC7426], the Application Plane includes both applications and services. In particular, the Application Plane incorporates the User Agent, a specialized application that interacts with the end user / operator and performs requests for Deterministic Networking services via an abstract Flow Management Entity, (FME) which may or may not be collocated with (one of) the end systems.

At the Application Plane, a management interface enables the negotiation of flows between end systems. An abstraction of the flow called a Traffic Specification (TSpec) provides the representation. This abstraction is used to place a reservation over the (Northbound) Service Interface and within the Application plane. It is associated with an abstraction of location, such as IP addresses and DNS names, to identify the end systems and eventually specify intermediate relay systems.

#### 4.2.2. The Controller Plane

The Controller Plane corresponds to the aggregation of the Control and Management Planes in [RFC7426], though Common Control and Measurement Plane (CCAMP) [CCAMP] makes an additional distinction between management and measurement. When the logical separation of the Control, Measurement and other Management entities is not relevant, the term Controller Plane is used for simplicity to represent them all, and the term controller refers to any device operating in that plane, whether is it a Path Computation entity or a Network Management entity (NME). The Path Computation Element (PCE) [PCE] is a core element of a controller, in charge of computing Deterministic paths to be applied in the Network Plane.

A (Northbound) Service Interface enables applications in the Application Plane to communicate with the entities in the Controller Plane.

One or more PCE(s) collaborate to implement the requests from the FME as Per-flow Per-Hop Behaviors installed in the relay systems for each individual flow. The PCEs place each flow along a deterministic sequence of relay systems so as to respect per-flow constraints such as security and latency, and optimize the overall result for metrics such as an abstract aggregated cost. The deterministic sequence can typically be more complex than a direct sequence and include redundancy path, with one or more packet replication and elimination points.

#### 4.2.3. The Network Plane

The Network Plane represents the network devices and protocols as a whole, regardless of the Layer at which the network devices operate.

The network Plane comprises the Network Interface Cards (NIC) in the end systems, which are typically IP hosts, and relay systems, which are typically IP routers and switches. Network-to-Network Interfaces such as used for Traffic Engineering path reservation in [RFC3209], as well as User-to-Network Interfaces (UNI) such as provided by the Local Management Interface (LMI) between network and end systems, are all part of the Network Plane.

A Southbound (Network) Interface enables the entities in the Controller Plane to communicate with devices in the Network Plane. This interface leverages and extends TEAS to describe the physical topology and resources in the Network Plane.

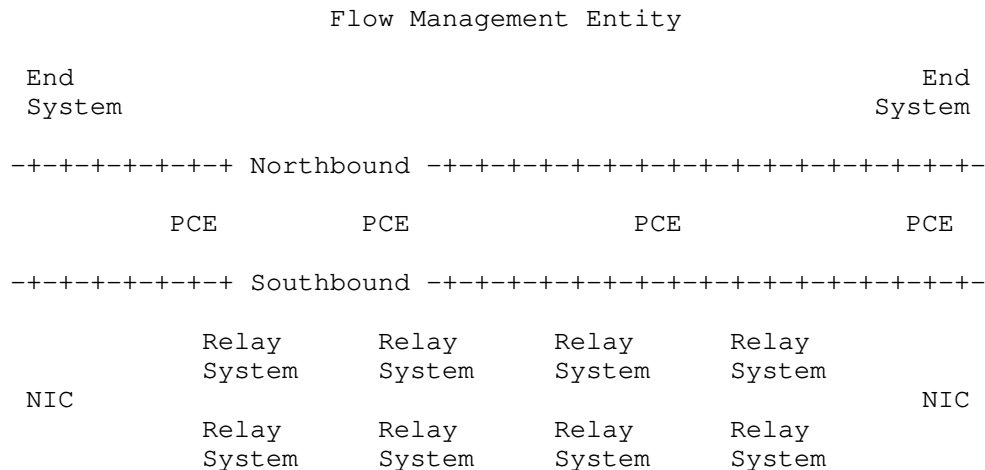


Figure 3

The relay systems (and eventually the end systems NIC) expose their capabilities and physical resources to the controller (the PCE), and update the PCE with their dynamic perception of the topology, across the Southbound Interface. In return, the PCE(s) set the per-flow paths up, providing a Flow Characterization that is more tightly coupled to the relay system Operation than a TSpec.

At the Network plane, relay systems exchange information regarding the state of the paths, between adjacent systems and eventually with the end systems, and forward packets within constraints associated to

each flow, or, when unable to do so, perform a last resort operation such as drop or declassify.

This specification focuses on the Southbound interface and the operation of the Network Plane.

#### 4.3. DetNet flows

##### 4.3.1. Source guarantees

DetNet flows can be synchronous or asynchronous. In synchronous DetNet flows, at least the relay systems (and possibly the end systems) are closely time synchronized, typically to better than 1 microsecond. By transmitting packets from different DetNet flows or classes of DetNet flows at different times, using repeating schedules synchronized among the relay systems, resources such as buffers and link bandwidth can be shared over the time domain among different DetNet flows. There is a tradeoff among techniques for synchronous DetNet flows between the burden of fine-grained scheduling and the benefit of reducing the required resources, especially buffer space.

In contrast, asynchronous DetNet flows are not coordinated with a fine-grained schedule, so relay and end systems must assume worst-case interference among DetNet flows contending for buffer resources. Asynchronous DetNet flows are characterized by:

- o A maximum packet size;
- o An observation interval; and
- o A maximum number of transmissions during that observation interval.

These parameters, together with knowledge of the protocol stack used (and thus the size of the various headers added to a packet), limit the number of bit times per observation interval that the DetNet flow can occupy the physical medium.

The source promises that these limits will not be exceeded. If the source transmits less data than this limit allows, the unused resources such as link bandwidth can be made available by the system to non-DetNet packets. However, making those resources available to DetNet packets in other DetNet flows would serve no purpose. Those other DetNet flows have their own dedicated resources, on the assumption that all DetNet flows can use all of their resources over a long period of time.

Note that there is no provision in DetNet for throttling DetNet flows (reducing the transmission rate via feedback); the assumption is that a DetNet flow, to be useful, must be delivered in its entirety. That is, while any useful application is written to expect a certain number of lost packets, the real-time applications of interest to DetNet demand that the loss of data due to the network is extraordinarily infrequent.

Although DetNet strives to minimize the changes required of an application to allow it to shift from a special-purpose digital network to an Internet Protocol network, one fundamental shift in the behavior of network applications is impossible to avoid--the reservation of resources before the application starts. In the first place, a network cannot deliver finite latency and practically zero packet loss to an arbitrarily high offered load. Secondly, achieving practically zero packet loss for unthrottled (though bandwidth limited) DetNet flows means that bridges and routers have to dedicate buffer resources to specific DetNet flows or to classes of DetNet flows. The requirements of each reservation have to be translated into the parameters that control each system's queuing, shaping, and scheduling functions and delivered to the hosts, bridges, and routers.

#### 4.3.2. Incomplete Networks

The presence in the network of relay systems that are not fully capable of offering DetNet services complicates the ability of the relay systems and/or controller to allocate resources, as extra buffering, and thus extra latency, must be allocated at points downstream from the non-DetNet relay system for a DetNet flow.

#### 4.4. Queuing, Shaping, Scheduling, and Preemption

As described above, DetNet achieves its aims by reserving bandwidth and buffer resources at every hop along the path of the DetNet flow. The reservation itself is not sufficient, however. Implementors and users of a number of proprietary and standard real-time networks have found that standards for specific data plane techniques are required to enable these assurances to be made in a multi-vendor network. The fundamental reason is that latency variation in one system results in the need for extra buffer space in the next-hop system(s), which in turn, increases the worst-case per-hop latency.

Standard queuing and transmission selection algorithms allow a central controller to compute the latency contribution of each relay node to the end-to-end latency, to compute the amount of buffer space required in each relay system for each incremental DetNet flow, and most importantly, to translate from a flow specification to a set of

values for the managed objects that control each relay or end system. The IEEE 802 has specified (and is specifying) a set of queuing, shaping, and scheduling algorithms that enable each relay system (bridge or router), and/or a central controller, to compute these values. These algorithms include:

- o A credit-based shaper [IEEE802.1Q-2014] Clause 34.
- o Time-gated queues governed by a rotating time schedule, synchronized among all relay nodes [IEEE802.1Qbv].
- o Synchronized double (or triple) buffers driven by synchronized time ticks. [IEEE802.1Qch].
- o Pre-emption of an Ethernet packet in transmission by a packet with a more stringent latency requirement, followed by the resumption of the preempted packet [IEEE802.1Qbu], [IEEE802.3br].

While these techniques are currently embedded in Ethernet and bridging standards, we can note that they are all, except perhaps for packet preemption, equally applicable to other media than Ethernet, and to routers as well as bridges.

#### 4.5. Coexistence with normal traffic

A DetNet network supports the dedication of a high proportion (e.g. 75%) of the network bandwidth to DetNet flows. But, no matter how much is dedicated for DetNet flows, it is a goal of DetNet to not interfere excessively with existing QoS schemes. It is also important that non-DetNet traffic not disrupt the DetNet flow, of course (see Section 4.6 and Section 7). For these reasons:

- o Bandwidth (transmission opportunities) not utilized by a DetNet flow are available to non-DetNet packets (though not to other DetNet flows).
- o DetNet flows can be shaped, in order to ensure that the highest-priority non-DetNet packet also is ensured a worst-case latency (at any given hop).
- o When transmission opportunities for DetNet flows are scheduled in detail, then the algorithm constructing the schedule should leave sufficient opportunities for non-DetNet packets to satisfy the needs of the uses of the network.

Ideally, the net effect of the presence of DetNet flows in a network on the non-DetNet packets is primarily a reduction in the available bandwidth.

#### 4.6. Fault Mitigation

One key to building robust real-time systems is to reduce the infinite variety of possible failures to a number that can be analyzed with reasonable confidence. DetNet aids in the process by providing filters and policers to detect DetNet packets received on the wrong interface, or at the wrong time, or in too great a volume, and to then take actions such as discarding the offending packet, shutting down the offending DetNet flow, or shutting down the offending interface.

It is also essential that filters and service remarking be employed at the network edge to prevent non-DetNet packets from being mistaken for DetNet packets, and thus impinging on the resources allocated to DetNet packets.

There exist techniques, at present and/or in various stages of standardization, that can perform these fault mitigation tasks that deliver a high probability that misbehaving systems will have zero impact on well-behaved DetNet flows, except of course, for the receiving interface(s) immediately downstream of the misbehaving device.

#### 4.7. Protocol Stack Model

[IEEE802.1CB], Annex C, offers a description of the TSN protocol stack. While this standard is a work in progress, a consensus around the basic architecture has formed. This stack is summarized in Figure 4.

## DetNet Protocol Stack

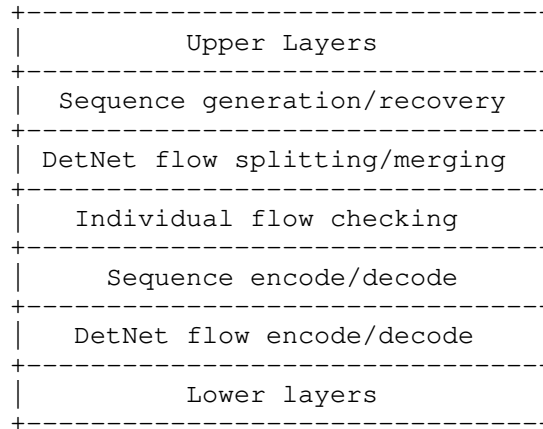


Figure 4

Not all layers are required for any given application, or even for any given network. The layers are, from top to bottom:

## Sequence generation/recovery

Supplies the sequence number for Seamless Redundancy (Section 3.3) for packets going down the stack, and discards duplicate packets coming up the stack.

## DetNet flow splitting/merging

Replicates packets going down the stack into two DetNet flows, and merges DetNet flows together for packets coming up the stack, based on the packet's DetNet flow identifier. Needed for Seamless Redundancy (Section 3.3).

## Individual flow checking

Examines packets belonging to individual flows, discards duplicate packets coming up the stack, and performs checks to detect contract violations.

## Sequence encode/decode

Encodes the sequence number into packets going down the stack, and extracts the sequence number from packets coming up the stack. This function may or may not be a null transformation of the packet, and for some protocols, is not explicitly present, being included in the DetNet flow encode/decode layer, below.

## DetNet flow encode/decode



Encapsulates packets going down the stack, based on the packet's locally-significant DetNet flow identifier, in order to identify to which DetNet flow the packet belongs, and extracts a locally-significant DetNet flow identifier from packets coming up the stack. This may be a null transformation (e.g., for DetNet flows identified by IP 5-tuple) or might be an explicit encapsulation (e.g., for DetNet flows identified with an MPLS label). DetNet flow identification is the basis for Seamless Redundancy, for assigning per-flow resources (if any) to packets and for defense against misbehaving systems (Section 4.6). When DetNet flows are assigned to pinned paths, this layer can be indistinguishable from the data forwarding layer(s).

The reader is likely to notice that Figure 4 does not specify the relationship between the DetNet layers, the IP layers, and the link layers. This is intentional, because they can usefully be placed different places in the stack, and even in multiple places, depending on where their peers are placed.

#### 4.8. Advertising resources, capabilities and adjacencies

There are three classes of information that a central controller needs to know that can only be obtained from the end systems and/or relay systems in the network. When using a peer-to-peer control plane, some of this information may be required by a system's neighbors in the network.

- o Details of the system's capabilities that are required in order to accurately allocate that system's resources, as well as other systems' resources. This includes, for example, which specific queuing and shaping algorithms are implemented (Section 4.4), the number of buffers dedicated for DetNet allocation, and the worst-case forwarding delay.
- o The dynamic state of an end or relay system's DetNet resources.
- o The identity of the system's neighbors, and the characteristics of the link(s) between the systems, including the length (in nanoseconds) of the link(s).

#### 4.9. Provisioning model

##### 4.9.1. Centralized Path Computation and Installation

A centralized routing model, such as provided with a PCE (RFC 4655 [RFC4655]), enables global and per-flow optimizations. (See

Section 4.2.) The model is attractive but a number of issues are left to be solved. In particular:

- o Whether and how the path computation can be installed by 1) an end device or 2) a Network Management entity,
- o And how the path is set up, either by installing state at each hop with a direct interaction between the forwarding device and the PCE, or along a path by injecting a source-routed request at one end of the path.

#### 4.9.2. Distributed Path Setup

Whether a distributed alternative without a PCE can be valuable should be studied as well. Such an alternative could for instance inherit from the Resource ReSerVation Protocol [RFC5127] (RSVP) flows.

In a Layer-2 only environment, or as part of a layered approach to a mixed environment, IEEE 802.1 also has work, either completed or in progress. [IEEE802.1Q-2014] Clause 35 describes SRP, a peer-to-peer protocol for Layer-2 roughly analogous to RSVP. Almost complete is [IEEE802.1Qca], which defines how ISIS can provide multiple disjoint paths or distribution trees. Also in progress is [IEEE802.1Qcc], which expands the capabilities of SRP.

#### 4.10. Scaling to larger networks

Reservations for individual DetNet flows require considerable state information in each relay system, especially when adequate fault mitigation (Section 4.6) is required. The DetNet data plane, in order to support larger numbers of DetNet flows, must support the aggregation of DetNet flows into tunnels, which themselves can be viewed by the relay systems' data planes largely as individual DetNet flows.

#### 4.11. Connected islands vs. networks

Given that users have deployed examples of the IEEE 802.1 TSN TG standards, which provide capabilities similar to DetNet, it is obvious to ask whether the IETF DetNet effort can be limited to providing Layer-2 tunnels between islands of bridged TSN networks. While this capability is certainly useful to some applications, and must not be precluded by DetNet, tunneling alone is not a sufficient goal for the DetNet WG. As shown in the Deterministic Networking Use Cases draft [I-D.grossman-detnet-use-cases], there are already deployments of Layer-2 TSN networks that are encountering the well-

known problems of over-large broadcast domains. Routed solutions, and combinations routed/bridged solutions, are both required.

## 5. Compatibility with Layer-2

Standards providing similar capabilities for bridged networks (only) have been and are being generated in the IEEE 802 LAN/MAN Standards Committee. The present architecture describes an abstract model that can be applicable both at Layer-2 and Layer-3, and over links not defined by IEEE 802. It is the intention of the authors (and hopefully, as this draft progresses, of the DetNet Working Group) that IETF and IEEE 802 will coordinate their work, via the participation of common individuals, liaisons, and other means, to maximize the compatibility of their outputs.

## 6. Open Questions

There are a number of architectural questions that will have to be resolved before this document can be submitted for publication. Aside from the obvious fact that this present draft is subject to change, there are specific questions to which the authors wish to direct the readers' attention.

### 6.1. Data plane shapers and schedulers

A number of techniques have been defined and are being defined by IEEE 802 for queuing, shaping, and scheduling transmissions on Ethernet media, most of which are directly applicable to any other medium. Specific selections of supported techniques are required, because minimizing, and even eliminating, congestion losses depends strongly on the details of the per-hop behavior of sources and relay systems.

The present authors expect that, at least, the IEEE 802 mechanisms will be supported.

### 6.2. DetNet flow identification and sequencing

The techniques to be used for DetNet flow identification must be settled. The following paragraphs provide a snapshot of the authors' opinions at the time of writing. These authors anticipate the submission of drafts in the near future on this subject.

IEEE 802.1 TSN streams are identified by giving each stream (DetNet flow) a {VLAN identifier, destination MAC address} pair that is unique in the bridged network, and that the MAC address must be a multicast address. If a source is generating, for example, two unicast UDP flows to the same destination, one DetNet and one not,

the DetNet flow's packets must be transformed at some point to have a multicast destination MAC address, and perhaps, a different VLAN than the non-DetNet flow's packets.

A similar provision would apply to DetNet packets that are identified by MPLS labels; any bridges between the LSRs need a {VLAN identifier, destination MAC address} pair uniquely identifying the DetNet flow in the bridged network.

Provision is made in current draft of [IEEE802.1CB] to make these transformations either in a Layer-2 shim in the source end system, on the output side of a router or LSR, or in a proxy function in the first-hop bridge. It remains to be seen whether this provision is adequate and/or acceptable to the IETF DetNet WG.

There are also questions regarding the sequentialization of packets for use with Seamless Redundancy (Section 3.3). [IEEE802.1CB] defines an Ethernet tag carrying a sequence number. If MPLS Pseudowires are used with a control word containing a sequence number, the relationship and interworking between these two formats must be defined.

### 6.3. Flat vs. hierarchical control

Boxes that are solely routers or solely bridges are rare in today's market. In a multi-tenant data center, multiple users' virtual Layer-2/Layer-3 topologies exist simultaneously, implemented on a network whose physical topology bears only accidental resemblance to the virtual topologies.

While the forwarding topology (the bridges and routers) are an important consideration for a DetNet Flow Management Entity (Section 4.2.1), so is the purely physical topology. Ultimately, the model used by the management entities is based on boxes, queues, and links. The authors hope that the work of the TEAS WG will help to clarify exactly what model parameters need to be traded between the relay systems and the controller(s).

### 6.4. Peer-to-peer reservation protocol

As described in Section 4.9.2, the DetNet WG needs to decide whether to support a peer-to-peer protocol for a source and a destination to reserve resources for a DetNet stream. Assuming that enabling the involvement of the source and/or destination is desirable (see Deterministic Networking Use Cases [I-D.grossman-detnet-use-cases]), it remains to decide whether the DetNet WG will make it possible to deploy at least some DetNet capabilities in a network using only a peer-to-peer protocol, without a central controller.

## 7. Security Considerations

Security in the context of Deterministic Networking has an added dimension; the time of delivery of a packet can be just as important as the contents of the packet, itself. A man-in-the-middle attack, for example, can impose, and then systematically adjust, additional delays into a link, and thus disrupt or subvert a real-time application without having to crack any encryption methods employed. See [RFC7384] for an exploration of this issue in a related context.

Furthermore, in a control system where millions of dollars of equipment, or even human lives, can be lost if the DetNet QoS is not delivered, one must consider not only simple equipment failures, where the box or wire instantly becomes perfectly silent, but bizarre errors such as can be caused by software failures. Because there is essential no limit to the kinds of failures that can occur, protecting against realistic equipment failures is indistinguishable, in most cases, from protecting against malicious behavior, whether accidental or intentional. See also Section 4.6.

Security must cover:

- o the protection of the signaling protocol
- o the authentication and authorization of the controlling systems
- o the identification and shaping of the DetNet flows

## 8. IANA Considerations

This document does not require an action from IANA.

## 9. Acknowledgements

The authors wish to thank Jouni Korhonen, Erik Nordmark, George Swallow, Rudy Klecka, Anca Zamfir, David Black, Thomas Watteyne, Shitanshu Shah, Craig Gunther, Rodney Cummings, Wilfried Steiner, Marcel Kiessling, Karl Weber, Ethan Grossman, Pat Thaler, and Lou Berger for their various contribution with this work.

## 10. Access to IEEE 802.1 documents

To access password protected IEEE 802.1 drafts, see the IETF IEEE 802.1 information page at <https://www.ietf.org/proceedings/52/slides/bridge-0/tsld003.htm>.

## 11. Informative References

- [AVnu] <http://www.avnu.org/>, "The AVnu Alliance tests and certifies devices for interoperability, providing a simple and reliable networking solution for AV network implementation based on the Audio Video Bridging (AVB) standards."
- [CCAMP] IETF, "Common Control and Measurement Plane", <<https://datatracker.ietf.org/doc/charter-ietf-ccamp/>>.
- [HART] [www.hartcomm.org](http://www.hartcomm.org), "Highway Addressable Remote Transducer, a group of specifications for industrial process and control devices administered by the HART Foundation".
- [HSR-PRP] IEC, "High availability seamless redundancy (HSR) is a further development of the PRP approach, although HSR functions primarily as a protocol for creating media redundancy while PRP, as described in the previous section, creates network redundancy. PRP and HSR are both described in the IEC 62439 3 standard.", <<http://webstore.iec.ch/webstore/webstore.nsf/artnum/046615!opendocument>>.
- [I-D.finn-detnet-problem-statement]  
Finn, N. and P. Thubert, "Deterministic Networking Problem Statement", draft-finn-detnet-problem-statement-04 (work in progress), October 2015.
- [I-D.grossman-detnet-use-cases]  
Grossman, E., Gunther, C., Thubert, P., Wetterwald, P., Raymond, J., Korhonen, J., Kaneko, Y., Das, S., and Y. Zha, "Deterministic Networking Use Cases", draft-grossman-detnet-use-cases-01 (work in progress), November 2015.
- [I-D.ietf-6tisch-architecture]  
Thubert, P., "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4", draft-ietf-6tisch-architecture-09 (work in progress), November 2015.
- [I-D.ietf-6tisch-tsch]  
Watteyne, T., Palattella, M., and L. Grieco, "Using IEEE802.15.4e TSCH in an IoT context: Overview, Problem Statement and Goals", draft-ietf-6tisch-tsch-06 (work in progress), March 2015.

- [I-D.ietf-roll-rpl-industrial-applicability]  
Phinney, T., Thubert, P., and R. Assimiti, "RPL applicability in industrial networks", draft-ietf-roll-rpl-industrial-applicability-02 (work in progress), October 2013.
- [I-D.svshah-tsvwg-deterministic-forwarding]  
Shah, S. and P. Thubert, "Deterministic Forwarding PHB", draft-svshah-tsvwg-deterministic-forwarding-04 (work in progress), August 2015.
- [IEEE802.1AS-2011]  
IEEE, "Timing and Synchronizations (IEEE 802.1AS-2011)", 2011, <<http://standards.ieee.org/getIEEE802/download/802.1AS-2011.pdf>>.
- [IEEE802.1BA-2011]  
IEEE, "AVB Systems (IEEE 802.1BA-2011)", 2011, <<http://standards.ieee.org/getIEEE802/download/802.1BA-2011.pdf>>.
- [IEEE802.1CB]  
IEEE, "Seamless Redundancy (IEEE Draft P802.1CB)", 2016, <<http://www.ieee802.org/1/files/private/cb-drafts/>>.
- [IEEE802.1Q-2014]  
IEEE, "MAC Bridges and VLANs (IEEE 802.1Q-2014)", 2014, <<http://standards.ieee.org/getIEEE802/download/802.1Q-2014.pdf>>.
- [IEEE802.1Qbu]  
IEEE, "Frame Preemption", 2016, <<http://www.ieee802.org/1/files/private/bu-drafts/>>.
- [IEEE802.1Qbv]  
IEEE, "Enhancements for Scheduled Traffic", 2016, <<http://www.ieee802.org/1/files/private/bv-drafts/>>.
- [IEEE802.1Qca]  
IEEE, "Path Control and Reservation", 2015, <<http://www.ieee802.org/1/files/private/ca-drafts/>>.
- [IEEE802.1Qcc]  
IEEE, "Stream Reservation Protocol (SRP) Enhancements and Performance Improvements", 2016, <<http://www.ieee802.org/1/files/private/cc-drafts/>>.

- [IEEE802.1Qch]  
IEEE, "Cyclic Queuing and Forwarding", 2016,  
<<http://www.ieee802.org/1/files/private/ch-drafts/>>.
- [IEEE802.1TSNTG]  
IEEE Standards Association, "IEEE 802.1 Time-Sensitive  
Networks Task Group", 2013,  
<<http://www.IEEE802.org/1/pages/avbridges.html>>.
- [IEEE802.3br]  
IEEE, "Interspersed Express Traffic", 2016,  
<<http://www.ieee802.org/3/br/>>.
- [IEEE802154]  
IEEE standard for Information Technology, "IEEE std.  
802.15.4, Part. 15.4: Wireless Medium Access Control (MAC)  
and Physical Layer (PHY) Specifications for Low-Rate  
Wireless Personal Area Networks", June 2011.
- [IEEE802154e]  
IEEE standard for Information Technology, "IEEE std.  
802.15.4e, Part. 15.4: Low-Rate Wireless Personal Area  
Networks (LR-WPANs) Amendment 1: MAC sublayer", April  
2012.
- [ISA100.11a]  
ISA/IEC, "ISA100.11a, Wireless Systems for Automation,  
also IEC 62734", 2011, <<http://www.isa100wci.org/en-US/Documents/PDF/3405-ISA100-WirelessSystems-Future-broch-WEB-ETSI.aspx>>.
- [ISA95]  
ANSI/ISA, "Enterprise-Control System Integration Part 1:  
Models and Terminology", 2000, <<https://www.isa.org/isa95/>>.
- [ODVA]  
<http://www.odva.org/>, "The organization that supports  
network technologies built on the Common Industrial  
Protocol (CIP) including EtherNet/IP."
- [PCE]  
IETF, "Path Computation Element",  
<<https://datatracker.ietf.org/doc/charter-ietf-pce/>>.
- [Profinet]  
<http://us.profinet.com/technology/profinet/>, "PROFINET is  
a standard for industrial networking in automation.",  
<<http://us.profinet.com/technology/profinet/>>.



- [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, <<http://www.rfc-editor.org/info/rfc2205>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<http://www.rfc-editor.org/info/rfc3209>>.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<http://www.rfc-editor.org/info/rfc4655>>.
- [RFC5127] Chan, K., Babiarz, J., and F. Baker, "Aggregation of Diffserv Service Classes", RFC 5127, DOI 10.17487/RFC5127, February 2008, <<http://www.rfc-editor.org/info/rfc5127>>.
- [RFC5673] Pister, K., Ed., Thubert, P., Ed., Dwars, S., and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks", RFC 5673, DOI 10.17487/RFC5673, October 2009, <<http://www.rfc-editor.org/info/rfc5673>>.
- [RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <<http://www.rfc-editor.org/info/rfc7384>>.
- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", RFC 7426, DOI 10.17487/RFC7426, January 2015, <<http://www.rfc-editor.org/info/rfc7426>>.
- [TEAS] IETF, "Traffic Engineering Architecture and Signaling", <<https://datatracker.ietf.org/doc/charter-ietf-teas/>>.
- [WirelessHART]  
www.hartcomm.org, "Industrial Communication Networks - Wireless Communication Network and Communication Profiles - WirelessHART - IEC 62591", 2010.

Authors' Addresses

Norman Finn  
Cisco Systems  
170 W Tasman Dr.  
San Jose, California 95134  
USA

Phone: +1 408 526 4495  
Email: nfinn@cisco.com

Pascal Thubert  
Cisco Systems  
Village d'Entreprises Green Side  
400, Avenue de Roumanille  
Batiment T3  
Biot - Sophia Antipolis 06410  
FRANCE

Phone: +33 4 97 23 26 34  
Email: pthubert@cisco.com

Michael Johas Teener  
Broadcom Corp.  
3151 Zanker Rd.  
San Jose, California 95134  
USA

Phone: +1 831 824 4228  
Email: MikeJT@broadcom.com

Internet Engineering Task Force  
Internet-Draft  
Intended status: Informational  
Expires: September 22, 2016

J. Huang, Ed.  
Huawei  
March 21, 2016

Integrated Mobile Fronthaul and Backhaul  
draft-huang-detnet-xhaul-00

Abstract

Ethernet can be a very promising technology for mobile Fronthaul and Backhaul traffic transportation, even an integrated Fronthaul / Backhaul (XHaul), although there are still some challenges. This memo tries to analyze some of the challenges and issues, such as L2 or L3 (MPLS/IP) forwarding, packet loss, etc., and to find out some 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 <http://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 September 22, 2016.

Copyright Notice

Copyright (c) 2016 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

1. Introduction . . . . .	3
1.1. Background . . . . .	3
1.2. Requirements Language . . . . .	3
1.3. Terminology . . . . .	3
2. Ethernet or MPLS or IP . . . . .	4
2.1. Scope . . . . .	4
2.2. Pinned Path . . . . .	4
2.3. QoS . . . . .	4
2.4. Protection . . . . .	5
2.5. Summary . . . . .	5
3. Encapsulation . . . . .	6
3.1. CPRI Aware or Unaware . . . . .	6
3.2. One Encapsulation over Multiple Technologies . . . . .	6
4. Packet Loss Due to BER . . . . .	6
5. Time Synchronization for Re-timing . . . . .	7
6. IANA Considerations . . . . .	8
7. Security Considerations . . . . .	8
8. References . . . . .	8
8.1. Normative References . . . . .	8
8.2. Informative References . . . . .	9
Author's Address . . . . .	10

## 1. Introduction

### 1.1. Background

5G network will be a heterogeneous network supporting "multiple types of access technologies" [NGMN-5G-WHITE-PAPER] . A network with very low latency and jitter to support these various access technologies can significantly increase network flexibility; and network slicing should be considered to separate technologies with different QoS requirements, and separate different operators, users or use cases. Ethernet is a good candidate for this purpose.

Fronthaul network has very critical delay, jitter and synchronization requirements, which is different from the existing Backhaul network. But in the future, there will be some new applications which require very low E2E latency, such as 5ms or even 1ms, as defined in [NGMN-5G-WHITE-PAPER] and [METIS-D1.1] . This will give some common requirements to both Fronthaul and Backhaul network.

There have been quite some work in the industry, trying to use Ethernet for Fronthaul, such as the IEEE 802.1CM and IEEE 1904.3.

Now the existing Backhaul network is Ethernet based (IP RAN, PTN, etc.), if the Fronthaul network can be Ethernet based too, it is possible to build an integrated Fronthaul and Backhaul

[XHaul] and [Crosshaul] are trying to develop and integrated Fronthaul and Backhaul, and packet network device ("Xhaul Packet Forwarding Element") will be considered. At the [IWPC-Evolving-Transport-Networks] meeting, some operators and vendors express the idea of "unified Fronthaul & Backhaul over Ethernet".

### 1.2. Requirements Language

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] .

### 1.3. Terminology

BBU: Baseband Unit

BER: Bit Error Rate

CPRI: Common Public Radio Interface

CRAN: Cloud / Centralized RAN

E2E: End to End

FEC: Forward Error Correction

FCS: Frame Check Sequence

FRR: Fast ReRoute

HARQ: Hybrid Automatic Repeat Request

IPWC: International Wireless Industry Consortium

RRU: Remote Radio Unit

TSN: Time Sensitive Network

## 2. Ethernet or MPLS or IP

### 2.1. Scope

The following analysis is on the Fronthaul / Backhaul use case.

MPLS includes L2VPN, L3VPN, PWE3, etc.

### 2.2. Pinned Path

If there are stringent QoS requirements, such as bandwidth reservation to avoid congestion, limited number of hops and distance to reduce delay, or even going through links with low BER, the path should be fixed. Traditional L2 MAC forwarding and L3 IP routing can not provided this capability. SDN or flow-based solution may be able to meet this requirement, either over L2 MAC forwarding or over L3 IP routing, in which forwarding decision will be made via MAC forwarding table, IP routing table or flow table which is installed by a controller, rather than being generated in an autonomous way, such as using OSPF protocol. But this type of solution is not yet widely used in the industrial. MPLS is a better solution for this purpose. A management system or PCE / RSVP / LDP can perform MPLS label planning and distribution, this is a mature solution in the industry.

### 2.3. QoS

In the Fronthaul / Backhaul use case, there will be Fronthaul traffic and Backhaul traffic in a same network, and also some other traffic types, such as WIFI, IoT traffic, etc. These various types of traffic have different QoS requirements. Priority based QoS mechanism is not sufficient. Pre-emption is developed by IEEE TSN to

resolve the interference from the low priority traffic. Besides, re-timing should also be considered to achieve very low jitter.

E2E resource reservation is necessary to avoid congestion. In a congestion case, the delay, jitter and packet loss will be a problem. Usually MPLS is used for E2E resource reservation.

If network slicing is considered to support various type of traffic in a network, and support multiple operator or tenants, traffic separation in the network is necessary. VLAN can serve this purpose in some common cases where bandwidth guarantee is not required. If the network will cover an area of a city, or a broader area, MPLS should be considered for E2E resource reservation and traffic separation. Multiple routing instances (such as OSPF) can be configured to serve this purpose, which usually work on port or port + VLAN.

#### 2.4. Protection

Protection is a common feature in operator's network.

Ethernet supports linear protection [ITU-G.8031] and ring protection [ITU-G.8032] , and a lot of other standard and proprietary solutions.

MPLS-TP can support multiple levels protection: LSP, PW and sector, linear protection [ITU-G.8131] . E2E resource reservation is retained after the protection switch.

Fast reroute is a MPLS (IP MPLS) [RFC4090] and IP [RFC5714] protection solution if there is link failure or router failure, which can provide network recovery within 50ms. The issue with IP fast reroute is, resource reservation can not be done via signaling, but has to depend on static network planning.

#### 2.5. Summary

Through the above analysis, MPLS (over Ethernet) is a good candidate for the XHaul case, mainly due to the E2E resource reservation and protection features. Support to MPLS should be considered.

But MPLS does not means it will work for all the case, e.g. in a pro-audio/video network, Ethernet may be a better choice because the network is small and simple, there are QoS requirements but not too stringent. It may be similar in the industry control network.

### 3. Encapsulation

#### 3.1. CPRI Aware or Unaware

[CPRI] is an open protocol, but it is not complete, some details are missing, such as the sampling bit width. Some possible values of sampling width are provided in the specification, but not sure which one will be used for a specific wireless technology, and if compression is considered to reduce bandwidth requirement, a smaller sampling width value may be used. If such a value is not specified, then it is difficult to identify a CPRI frame.

A reasonable solution is to treat the CPRI traffic as a bit stream. A fixed block of CPRI traffic, such as 1500byte including the encapsulation, or the traffic over a fixed period of time, is encapsulate into a packet.

One of the advantages of CPRI aware encapsulation, is to perform compression, and some of the reserved bits in the control bit are removed, the IQ data is compressed using some compressing solution. But, maybe the RRU itself is a better place to do this kind of processing, rather than in the transport device.

#### 3.2. One Encapsulation over Multiple Technologies

IEEE 1904.3 defines encapsulation for CPRI over Ethernet. Should that encapsulation format be used over MPLS or even IP too, or should there be any necessary changes?

### 4. Packet Loss Due to BER

The CPRI traffic carries the IQ data of baseband signal, in which FEC function is usually used, e.g. the turbo coding function in LTE. Some bit errors in the baseband signal or in the IQ data can be corrected by the FEC module, and the left can be fixed using HARQ retransmission mechanism. Due to this, when CPRI traffic is carried by direct fiber link or by non-packet based technology, such as OTN, even if there are bit errors, it is not a big problem, the BBU can still process the IQ data.

But if CPRI traffic is carried over an Ethernet or other packet-based link, when there is a bit error, usually the packet is discarded. The packet size will decide how much CPRI traffic be lost. Because CPRI requires a lot of bandwidth, the packet size should be large enough for efficiency. For Ethernet the payload size should be 1500byte or 9000byte (jumbo frame). If such a continuous segment of CPRI data is lost, it will significantly increases "equivalent" BER



[packet-loss-consideration] . Issues caused by packet loss can not be fixed by existing FEC function in LTE. HARQ retransmission will have to be considered as a final resort.

The packetization / framing will make the issue worse. The CPRI traffic over a packet may expand across multiple CPRI basic frame or even hyperframe, and further across multiple LTE code block and transport block, which may lead to multiple LTE HARQ retransmission. Further study on the impact to the wireless network performance caused by packet loss is necessary.

Cut-through forwarding is to start forwarding actions such as forwarding table lookup when the header of a packet is received, before receiving the complete packet. Cut-through forwarding can significantly reduce the delay in a network device. Receiving a packet of 1500bytes on a 10GE interface will take about 1.2us, if cut-through forwarding is used, more than 1us delay time can be reduced. Cut-through forwarding is widely used in FCoE and Infiniband, some Ethernet switches also provide this capability.

But cut-through forwarding has some issues, one of which is the FCS error of an Ethernet packet. If there is a bit error, the FCS validation will fail, and the packet should be discarded. But in cut-through forwarding mode, before the switch can validate the FCS, part of the packet is already on the wire and the packet can not be discarded. The packet with bit error will finally be forwarded to a store-and-forward switch, or the final receiver. Even the receiver, such as the BBU in the CRAN architecture, finally receives the packet, it will have to discard the packet, because it does not know the bit error occurs in which part of the packet, in the Ethernet or MPLS header, or the encapsulation, or in the CPRI data.

Cut-through forwarding itself does not help in the bit error case.

## 5. Time Synchronization for Re-timing

Due to the very critical jitter requirement, +/- 8.138ns for one way jitter and +/- 16.276ns for round trip jitter, it is very difficult to achieve this target simply via scheduling, neither priority based nor pre-emption, or other algorithms. According to [applicability-of-qbu-and-qbv], even if pre-emption is used, a maximum delay of 114.4ns over a 10GE interface still exists. To further reduce the jitter, re-timing should be used. That is, put a time stamp T1 in the packet at the ingress of the network; when the packet arrives at the egress node at T2, buffer the packet until T3, then send out the packet.  $T3 \geq T2$ .  $(T3 - T1)$  is a fixed value, and it should be long enough to cover all the possible jitter, fibre

propagation delay, processing delay, etc. On the other hand, the delay ( $T3-T1$ ) should be as low as possible.

Re-timing mechanism should be bi-directional.

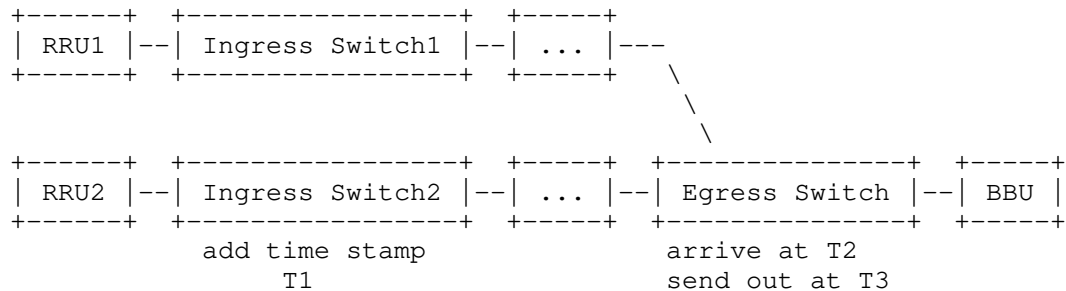


Figure 1

The re-timing mechanism depends on accuracy of time synchronization at the ingress nodes and the egress nodes. In a common scenario, in time synchronization a network device will trace to its uplink network device, such as the ingress switch will trace to the egress switch as shown in the above figure. The time alignment error (TAE) between the ingress switch and the egress switch may impact the delay ( $T3-T1$ ) if TAE is variable over the time. The variation of TAE over the time must be small enough so as to minimize jitter; if the TAE is a fixed value over the time, it is not a problem. The detail requirement needs further study.

## 6. IANA Considerations

This memo includes no request to IANA.

## 7. Security Considerations

TBD.

## 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, <<http://www.rfc-editor.org/info/rfc2119>>.

## 8.2. Informative References

- [CPRI] CPRI Alliance, "CPRI Specification V7.0", 2015.
- [Crosshaul] 5G-PPP, "5G-Crosshaul: The 5G Integrated fronthaul/backhaul transport network", 2015, <<https://5g-ppp.eu/xhaul/>>.
- [ITU-G.8031] ITU, "G.8031 : Ethernet linear protection switching", 2015, <<https://www.itu.int/rec/T-REC-G.8031-201501-I/en>>.
- [ITU-G.8032] ITU, "G.8032 : Ethernet ring protection switching", 2014, <<https://www.itu.int/rec/T-REC-G.8032-201508-I/en>>.
- [ITU-G.8131] ITU, "G.8131 : Linear protection switching for MPLS transport profile", 2014, <<https://www.itu.int/rec/T-REC-G.8131-201407-I/en>>.
- [IWPC-Evolving-Transport-Networks] IWPC, "Evolving Transport Networks", 2016, <<http://www.iwpc.org/ResearchLibrary.aspx?ArchiveID=234&Display=doc>>.
- [METIS-D1.1] METIS, "Deliverable D1.1 Scenarios, requirements and KPIs for 5G mobile and wireless system", 2013.
- [NGMN-5G-WHITE-PAPER] NGMN Alliance, "NGMN-5G-White-PAPER", 2015, <[https://www.ngmn.org/uploads/media/NGMN\\_5G\\_White\\_Paper\\_V1\\_0.pdf](https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf)>.
- [RFC4090] Pan, P., Ed., Swallow, G., Ed., and A. Atlas, Ed., "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, DOI 10.17487/RFC4090, May 2005, <<http://www.rfc-editor.org/info/rfc4090>>.
- [RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, DOI 10.17487/RFC5714, January 2010, <<http://www.rfc-editor.org/info/rfc5714>>.
- [XHaul] 5G-PPP, "5G-XHaul: Dynamically Reconfigurable Optical-Wireless Backhaul/Fronthaul with Cognitive Control Plane for Small Cells and Cloud-RANs", 2015, <<https://5g-ppp.eu/5g-xhaul/>>.

[applicability-of-qbu-and-qbv]

Farkas, J. and B. Varga, "Applicability of Qbu and Qbv to Fronthaul", 2015, <<http://www.ieee802.org/1/files/public/docs2015/cm-farkas-applicability-of-bu-and-bv-1115-v02.pdf>>.

[packet-loss-consideration]

Varga, B. and J. Farkas, "Packet/Frame loss considerations for CPRI over Ethernet", 2016, <<http://www.ieee802.org/1/files/public/docs2016/cm-varga-CPRI-packetloss-considerations-0116-v02.pdf>>.

#### Author's Address

James Huang (editor)  
Huawei  
Shenzhen,  
China

Phone:  
Email: [james.huang@huawei.com](mailto:james.huang@huawei.com)



Network Working Group  
Internet Draft  
Intended status: Informational  
Expires: September 2016

Y. Zha  
Huawei Technologies  
L. Geng  
China Mobile

March 16, 2016

Deterministic Networking Requirements on Data and Control Plane  
draft-zha-detnet-requirements-00

Abstract

Deterministic Networking (DetNet) is focused on how to serve time critical flow with low data loss and bounded delay. Unlike contemporary solution which improves QoS such as TE, redundant bandwidth provisioning and dedicated channel reservation, DetNet provides more general approaches that use IP-based techniques and guarantee the worst-case latency of DetNet flows while allowing sharing among best-effort flows. For this purpose, DetNet may require upgraded or redefined data plane as well as control plane, since current networking cannot assure maximum end-to-end latency. This document describes some technical requirements on possible data plane, control plane and DetNet flow modeling that can help to clarify those capabilities DetNet should have.

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), 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/ietf/lid-abstracts.txt>

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

This Internet-Draft will expire on September 16, 2016.

#### Copyright Notice

Copyright (c) 2014 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

1. Introduction .....	2
2. Conventions used in this document .....	3
3. Overall Architecture .....	4
4. Data Plane Requirements .....	4
4.1. MPLS to Support DetNet Flow Forwarding .....	4
4.2. DetNet Flow Identification .....	5
4.3. Deterministic Forwarding Capability .....	6
5. Control Plane Requirements .....	7
5.1. Distributed or Centralized Control .....	7
5.2. Southbound/Northbound Interfaces .....	8
5.3. Peer-to-Peer Reservation Protocol .....	9
6. Requirements on DetNet Flow Modeling .....	9
7. Requirements on Synchronization and OAM .....	10
8. Security Considerations .....	10
9. IANA Considerations .....	10
10. Acknowledgments .....	10
11. References .....	10
11.1. Normative References .....	10
11.2. Informative References .....	11

#### 1. Introduction

The rapid growth of the existing communication system and its access into almost all aspects of daily life has led to great dependency on services it provides. The communication network, as it is today, has applications such as multimedia and peer-to-peer file sharing distribution that require Quality of Service (QoS) which guarantees delay and jitter to maintain a certain level of

performance. Meanwhile, cellular network has become key element in modern social network with increasing popularity over the past years. A communication system with extreme real-time and high reliability is essential for the next concurrent and next generation cellular networks as well as its bearer network for E-2-E performance requirements.

Conventional transport network is IP-based for the benefits of high bandwidth and low cost. However, the delay and jitter guarantee becomes a challenge in case of contention since the service is not deterministic but best effort. With more and more rigid demand in latency control in the future network [METIS], deterministic networking [I-D.finn-detnet-architecture] becomes a promising solution to meet the requirement of ultra low-latency of certain applications and use cases. There are already typical issues for latency sensitive networking requirements in midhaul and backhaul network to support LTE and future 5G network [5G]. Moreover not only the telecom industry but also other vertical industries have increasing demand on delay-sensitive communications as automation becomes increasingly popular recently.

More specifically, CoMP techniques, D-2-D, industrial automation and gaming/media service all have great dependency on low delay communications as well as high reliability to guarantee the service performance. Note that the deterministic networking is not equal to low latency as it is more focused on the worst case delay bound of the duration of certain application or service. It can be argued that without high certainty and absolute delay guarantee, low delay provisioning is just relative [RFC3393], which is not sufficient to some delay-critical service since delay violation in an instance cannot be tolerated. Overall, the requirements from vertical industries seem to be well aligned with the expected low latency and high determinist performance of future networks

This document only describes technical requirements and design principles on data plane, control plane and modeling to minimize the scope of this document since a full coverage of DetNet can be found in [I-D.finn-detnet-architecture]

## 2. Conventions used in this document

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]. In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying [RFC2119] significance.



### 3. Overall Architecture

Draft [I-D.finn-detnet-architecture] provides an overall picture of the DetNet operation. The draft covers almost every aspects of deterministic service provisioning which will be indeed asking for an architectural upgrade. An entire working system is needed for end-to-end delay-guaranteed service provisioning, but the WG may only works on several topics as the final deliverables. Meanwhile, there are still some open issues on both design and implementation of data plane and control plane.

Based on the current architecture, three key aspects are identified: how to specify the data plane (including the forwarding in [I-D.finn-detnet-architecture]) and what is the essential characteristic of the data plane enabling deterministic forwarding; how to design appropriate control protocols and mechanisms to serve end-to-end DetNet flows; what and how modeling should be defined to describe basic principles of both data plane and control plane.

### 4. Data Plane Requirements

As mentioned in the current charter, DetNet focus on Layer 3 techniques to support deterministic latency applications. Without redefining the hardware and physical layers, potential data plane must be compatible with Layer 2 operations (i.e. TSN) In the architectural draft[I-D.finn-detnet-architecture], the data plane is described as the most fundamental element of the network, comprising the device and forwarding plane, which decides the queuing, scheduling and shaping of traffic. In order to provide converged networking in which DetNet flows get bounded delay guarantee while non-DetNet flows are best-effort, the DetNet data plane should specify to answer the following questions: how to use existing technologies for Layer 3 operation, MPLS may be a good candidate to setup end-to-end deterministic flow path; how to identify the DetNet data flow; a standardized mechanism of DetNet flow forwarding including queuing, transmission, shaping and etc.

#### 4.1. MPLS to Support DetNet Flow Forwarding

IEEE works on Layer 1-2 technologies where TSN has successfully provided a potential solution to the problem of serving timing critical data flow in Ethernet. In order to expand the guaranteed delay provisioning to the scale of the entire network, it is critical for DetNet WG to define the Layer forwarding. As mentioned in current charter, the goal is to use existing

technology that can be compatible with TSN work. Given that, MPLS technology is a good candidate for its maturity and robustness.

MPLS lies between Layer 3 and 2 with an encapsulation of different protocols. Supposing that deterministic forwarding is ready at each hop under TSN, MPLS can carry Ethernet frames to utilize TSN techniques to provide an end-to-end label switched DetNet path. In this way, the WG may need to further define how to use MPLS in the following aspects:

- Setup Label Switched Path (LSP) for DetNet flow with label definition and QoS mapping.

- Latency-aware LSP installation and removing. RSVP-TE may not be suitable for DetNet flows as it only describes macro-parameters such as bandwidth. An extension may be desirable with latency parameter.

- Supporting Layer-2 techniques such as pseudowire.

#### 4.2. DetNet Flow Identification

Identification is the first step among all corresponding operations related to DetNet flows. Because of the unified transmission of both deterministic traffic flow and conventional best effort flows, DetNet flow needs to be identified for certain processes. Basically, there are two questions that need to be taken care of based on the authors' opinions: first, how to identify the flow entity; second, how to identify its delay bound requirement.

Flow identification can be done via some tuple matching approach, such as {VLAN identifier, destination MAC address} pair, source/destination address, port, MAC address and so on. All these approaches can mark a unique flow in the network. However, they are all dependent on the source. In other words, if the source is sending both DetNet flow and best effort flow, this approach does not work. To tackle with this challenge, a different destination MAC address or a VLAN ID could be used. And so does the MPLS labels.

A transformation in Layer 2 or a proxy function could be a solution. More fundamental solution can be redefining new flag or defining new flow model without doing the tuple matching. Meanwhile, the flow identification should be application-aware and independent of sending source. An alternative approach can be similar to those in Service Function Chain (SFC) to define encapsulation format be agnostic to the layer at which it is

applied and the service that is being constructed. However, this provision certainly need more work depending on the acceptance of WG.

The second issue is how to identify the delay bound of the target flow being required. This is a new question which has not been well discussed. For DetNet, we want to do absolute delay bound provisioning while allowing best effort traffic to share unscheduled traffic. In this way, how to provision just enough resource to the DetNet flow is the major problem. So first needs is to know the delay bound of the application flow. Treat all DetNet flow as the same does not make sense since one flow requires 1ms delay but the other requires 10ms which are all deterministic but definitely needs different amount of network resources.

Currently, the flow can be marked with QoS level, e.g. 0-7, denotes the level of priority of the flow. There is no absolute delay information of the flow as QoS level only specifies the general characteristic of the data flow. A new mechanism to label the absolute delay bound is necessary and it can be done via a mapping algorithm between delay bound and current QoS labeling.

#### 4.3. Deterministic Forwarding Capability

Although, DetNet will not define new hardware or physical layer, different flow forwarding mechanism including queuing, transmission selection, and shaping equipped by different network devices or network elements can lead to the uncertainty of the timing. To achieve predictable delay introduced in every hop, a standard of queuing, transmission selection, shaping, and preemption is needed to unify all the NEs in forwarding plane. TSN can be expected to support this but sufficiently well-defined characteristics should be defined in the WG. In the authors' opinion, at least two features should be defined:

-Frame preemption. As transmitted via the same medium, deterministic flow should always have absolute priority against best effort flow, which means the DetNet flow should be first scheduled if there is any. The problem is if the port is transmitting a best effort flow, the incoming DetNet flow has to wait a MTU transmission time thus introduce 12us delay in a 1GE line. So the best effort flow should be permeable to the DetNet flow. TSN 802.1qbu is working preemption solution for Ethernet, DetNet should be expected to have this capability either in Layer 2 or Layer 3. In addition to current TSN solution which is mainly focused on cut best effort packet into smaller packet with

encapsulation, preemption on demand or real time preemption should also be considered to avoid overhead and save latency.

-Time aware scheduling: preemption can solve the conflict between DetNet flow and best effort flow by giving DetNet flow absolute priority to preempt normal traffic. Conflict between DetNet flows can still introduce uncertain delay. So scheduling of DetNet traffic in advance with time aware capability is desirable to solve this conflict.

## 5. Control Plane Requirements

In the use case draft [I-D.draft-ietf-detnet-use-cases], several use cases summarize the needs of unified control and management protocols and control plane. Control plane is the key component of DetNet that can unify the existing Layer 2 technology and Layer 3 to deliver a fully functional solution for delay guarantee service. Note that here "control plane" is used just for simplicity to represent all control and management mechanism and protocols which does not mean the separation of control and forwarding plane in SDN.

The DetNet control plane design should include: architectural choice as distributed or centralized control; southbound/northbound interface; peer-to-peer reservation protocols.

### 5.1. Distributed or Centralized Control

Assuming the data plane upgrade can provide deterministic forwarding behavior at each hop, a unified control mechanism is demanded to provide end-to-end delay guarantee. Although the architecture draft propose a controller based centralized control plane mechanism, it has not been decided yet to solely focus on centralized only. The WG should also consider some distributed solution with reservation protocols since centralized resource reservation may introduce addition latency.

Introducing centralized controller seems to be the simple and stringent forward solution. SDN is the new fashion right now with separation of control plane from device to a remote controller. For DetNet, if there is a central controller can do the path computing, resource reservation and transmission selection based on the information and delay requirement of the target flow on every hop, it definitely can help to minimize the delay. First of all, path computing has to be relied on central controller to select the best forwarding path based on the DetNet flow request

and global information of the network. Secondly, reserve enough resource at every hop is challenging for end-to-end delay guarantee which can be benefited from a control resource manager to make the DetNet flow get just enough resource at every hop. Thirdly, transmission selection or scheduling at each hop is also crucial to serve the flow in time. A central arbiter can be equipped to make the DetNet flow travel pass the multi-hop with green light all the way

On the other hand, centralized controller is offsite and has to communicate with all the NEs of entire network which can bring in addition latency. For DetNet flows that require ms delay, the time cost of communication to the controller and communication between control and NEs is not affordable. So a distributed control mechanism is also considered to be promising in some scenarios. With distributed control, the DetNet flows do not have to wait the controller to acknowledge and configure the downstream NEs. An efficient reserve protocol is needed to reserve bandwidth, buffer and other resource at each hop along the forwarding path. Also, a hybrid approach can also be helpful as the controller can setup the path and distributed protocol to reserve the resource at each hop.

## 5.2. Southbound/Northbound Interfaces

Within SDN architecture, southbound and northbound are the key interfaces which are close related the NEs and flow model. If there is a central controller for DetNet, it needs to know the forwarding capabilities of the NEs and then send the configuration to the NEs to serve the DetNet flow. All of this information is transmitted via southbound interface. Also the controller should get the service level requirements via northbound interface which is based on the service model of DetNet flows.

Southbound interface should enables communication between control plane and network elements with following information:

- The resource inventory of the network elements, such as left over bandwidth, unscheduled time slot on link or the size of the unused buffer.
- Topology information of the network, it can be the similar work that is being done in I2RS or TEAS.
- The data plane information of NEs, which include the queuing, transmission selection, shaping and preemption related information.

- The scheduling or QoS setting on the data plane and the configuration information that makes the change.

Northbound interface enables communication between applications and control and it should contain information related to:

- Service level delay requirement which can be transformed into device configuration change in the controller.

- Flow and service description which can be relied on flow model and configuration model.

### 5.3. Peer-to-Peer Reservation Protocol

As mentioned in section 5.1, distributed reservation protocols are also desired even in a centralized architecture to reduce the setup time caused by communication with controller. And ideal case is that, a peer-to-peer protocol for a S-D pair to reserve resource for DetNet flow without a central controller. Of course this will be an efficient and sophisticated approach if WG can make it possible to deploy. In the authors' opinion, this peer-to-peer reservation protocol should have characteristics such as:

- A hop by hop reservation protocol that reserve resource for the coming DetNet flow before arriving to the next hop. The DetNet flow will just pass through the hop using pre-setup resource.

- Only control packet or reservation signal is processed at each hop, DetNet flow will be transmitted transparently all the way to destination.

- This multi-hop signaling should start transmission of DetNet flow immediately after setting up the path to next hop without configure end-to-end path.

## 6. Requirements on DetNet Flow Modeling

DetNet flow modeling is one the most important deliverables of this WG. How to model the DetNet flow will decide how to serve the flow and how to reserve the resource, which are all the main focus of the WG. Model is about description of characteristic of flow transmission, technical requirements and network resource demands. Traditional flow model is based on RSVP model which includes peak rate, sustain rate, burst and etc. this is not feasible for deterministic service provisioning as the bandwidth itself is not an accurate description of latency. The DetNet flow modeling should include:

- Application-aware description of the flow.
- Timing information of the flow so that the network can provide accurate service to guarantee the delay requirement.
- Data transmission information of the flow including packet size, interval, traffic pattern and so on.
- Network-aware constraints on networking environment.

## 7. Requirements on Synchronization and OAM

Since operations and scheduling can be time-aware in DetNet and absolute delay bound is a must in a multi-hop network, time synchronization is necessary. Besides, in both centralized and distributed architecture, delay measuring among multi-hops and synchronization is a must for DetNet.

There is also a need for OAM system and protocols which can help to provide E2E delay sensitive service provisioning.

More details of synchronization and OAM will be provided in next version.

## 8. Security Considerations

TBD

## 9. IANA Considerations

This document has no actions for IANA.

## 10. Acknowledgments

This document has benefited from reviews, suggestions, comments and proposed text provided by the following members, listed in alphabetical order: Yuanlong Jiang and Oilver Huang.

## 11. References

### 11.1. Normative References

- [RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3393] C. Demichelis, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM) ", RFC 3393, November 2002.

## 11.2. Informative References

[I-D.finn-detnet-problem-statement]

Finn, N. and P. Thubert, "Deterministic Networking Problem Statement", draft-finn-detnet-problem-statement-02 (work in progress), October 2015.

[I-D.finn-detnet-architecture]

Finn, N., Thubert, P., and M. Teener, "Deterministic Networking Architecture", draft-finn-detnet-architecture-03 (work in progress), March 2016.

[I-D.draft-ietf-detnet-use-cases]

Ethan Grossman, et al, "Deterministic Networking Use Cases", draft-ietf-detnet-use-cases-08, March 2016.

[METIS] METIS Document Number: ICT-317669-METIS/D1.1, Scenarios, requirements and KPIs for 5G mobile and wireless system, April 29, 2013. Available on line at: <[https://www.metis2020.com/wp-content/uploads/deliverables/METIS\\_D1.1\\_v1.pdf](https://www.metis2020.com/wp-content/uploads/deliverables/METIS_D1.1_v1.pdf)>

[5G] Ericsson white paper, "5G Radio Access, Challenges for 2020 and Beyond." June 2013. Available at:  
<<http://www.ericsson.com/res/docs/whitepapers/wp-5g.pdf>>

[CoMP] NGMN Alliance, "RAN EVOLUTION PROJECT COMP EVALUATION AND ENHANCEMENT ", MARCH 2015,  
<[https://www.ngmn.org/uploads/media/NGMN\\_RANEV\\_D3\\_CoMP\\_Evaluation\\_and\\_Enhancement\\_v2.0.pdf](https://www.ngmn.org/uploads/media/NGMN_RANEV_D3_CoMP_Evaluation_and_Enhancement_v2.0.pdf)>

[LTE-Latency] Samuel Johnston, "LTE Latency: How does it compare to other technologies?" report of OpenSignal March 10, 2014.  
<<http://opensignal.com/blog/2014/03/10/lte-latency-how-does-it-compare-to-other-technologies/>>

[EA12] P. C. Evans, M. Annunziata, "Industrial Internet: Pushing the Boundaries of Minds and Machines", General Electric White paper, November 2012.



[UHD-video] Petr Holub, "Ultra-High Definition Videos and Their Applications over the Network", The 7th International Symposium on VICTORIES Project, OCTOBER 8, 2014. <[http://www.aist-victories.org/jp/7th\\_sympo\\_ws/PetrHolub\\_presentation.pdf](http://www.aist-victories.org/jp/7th_sympo_ws/PetrHolub_presentation.pdf)>

#### Authors' Addresses

Yiyong Zha  
Huawei Technologies  
Email: zhayiyong@huawei.com

Liang Geng  
China Mobile  
Email: liang.geng@hotmail.com

