Workgroup: Network Working Group
Internet-Draft: draft-ietf-detnet-yang-13
Published: 12 May 2021
Intended Status: Standards Track
Expires: 13 November 2021
Authors:
X. Geng, *Huawei Technologies*
M. Chen, *Huawei Technologies*
Y. Ryoo, *ETRI*
D. Fedyk, *LabN Consulting, L.L.C.*
R. Rahman, *Individual*
Z. Li, *China Mobile*

Deterministic Networking (DetNet) YANG Model

Abstract

This document contains the specification for the Deterministic Networking YANG Model for configuration and operational data for DetNet Flows. The model allows for provisioning of end-to-end DetNet service along the path without dependency on any signaling protocol. It also specifies operational status for flows.

The YANG module defined in this document conforms to the Network Management Datastore Architecture (NMDA).

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].

Status of This Memo

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

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

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

This Internet-Draft will expire on 13 November 2021.

Copyright Notice

Table of Contents

1. Introduction

DetNet (Deterministic Networking) provides a capability to carry specified unicast or multicast data flows for real-time applications with extremely low packet loss rates and assured maximum end-to-end delivery latency. A description of the general background and concepts of DetNet can be found in [RFC8655].

This document defines a YANG model for DetNet based on YANG data types and modeling language defined in [RFC6991] and [RFC7950]. DetNet service, which is designed for describing the characteristics of services being provided for application flows over a network, and DetNet configuration, which is designed for DetNet flow path establishment, flow status reporting, and DetNet functions configuration in order to achieve end-to-end bounded latency and zero congestion loss, are both included in this document.

2. Terminology

This document uses the terminology defined in [RFC8655].

3. DetNet YANG Module

The DetNet YANG module includes DetNet App-flow, DetNet Service Sub-layer, and DetNet Forwarding Sub-layer configuration and operational objects. The corresponding attributes used in different sub-layers are defined in Section 3.1, 3.2, 3.3 respectively.

3.1. DetNet Application Flow YANG Attributes

DetNet application flow is responsible for mapping between application flows and DetNet flows at the edge node(egress/ingress node). The the application flows can be either layer 2 or layer 3 flows. To map a flow at the User Network Interface (UNI), the corresponding attributes are defined in [I-D.ietf-detnet-flow-information-model].

3.2. DetNet Service Sub-layer YANG Attributes

DetNet service functions, e.g., DetNet tunnel initialization/termination and service protection, are provided in the DetNet service sub-layer. To support these functions, the following service attributes need to be configured:

- DetNet flow identification
- Service function indication, indicates which service function will be invoked at a DetNet edge, relay node or end station. (DetNet tunnel initialization or termination are default functions in DetNet service layer, so there is no need for explicit indication). The corresponding arguments for service functions also needs to be defined.

3.3. DetNet Forwarding Sub-layer YANG Attributes

As defined in [RFC8655], DetNet forwarding sub-layer optionally provides congestion protection for DetNet flows over paths provided by the underlying network. Explicit route is another mechanism that is used by DetNet to avoid temporary interruptions caused by the convergence of routing or bridging protocols, and it is also implemented at the DetNet forwarding sub-layer.

To support congestion protection and explicit route, the following transport layer related attributes are necessary:

- Flow Specification and Traffic Requirements, refers to [I-D.ietf-detnet-flow-information-model]. These may used for resource reservation, flow shaping, filtering and policing by a control plane or other network management and control mechanisms.
- Since this model programs the data plane existing explicit route mechanisms can be reused. If a static MPLS tunnel is used as the transport tunnel, the configuration need to be at every transit node along the path. For an IP based path, the static configuration is similar to the static MPLS case. This document provides data-plane configuration of IP addresses or MPLS labels but it does not provide control plane mapping or other aspects.

4. DetNet Flow Aggregation

DetNet provides the capability of flow aggregation to improve scalability of DetNet data, management and control planes. Aggregated flows can be viewed by the some DetNet nodes as individual DetNet flows. When aggregating DetNet flows, the flows should be compatible: if bandwidth reservations are used, the reservation should be a reasonable representation of the individual reservations; if maximum delay bounds are used, the system should ensure that the aggregate does not exceed the delay bounds of the individual flows.

The DetNet YANG model defined in this document supports DetNet flow aggregation with the following functions:

- Aggregation flow encapsulation/decapsulation/identification
- Mapping individual DetNet flows to an aggregated flow

- Changing traffic specification parameters for aggregated flow

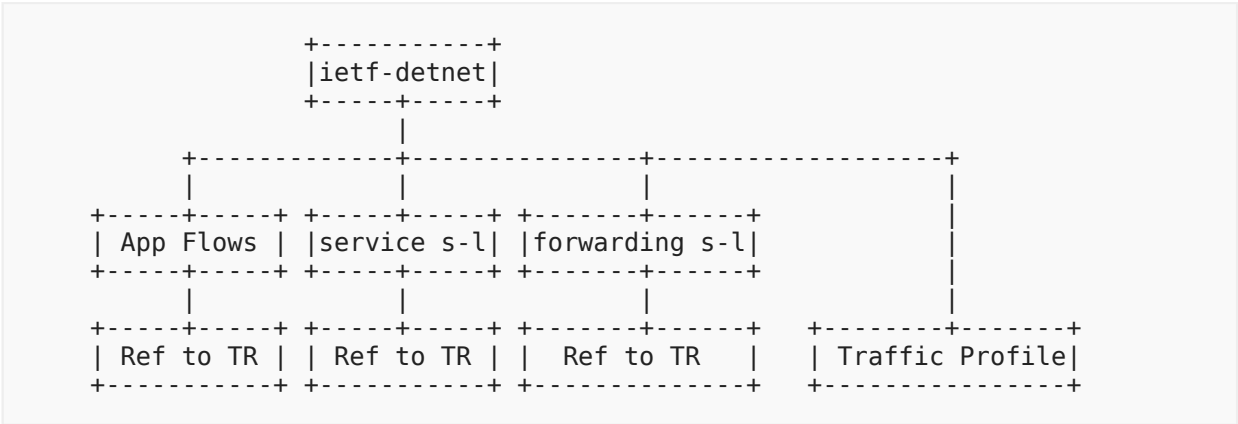
The following cases of DetNet aggregation are supported:

- Ingress node aggregates App flows into a service sub-layer of DetNet flow
- In ingress node, the service sub-layers of DetNet flows are aggregated into a forwarding sub-layer
- In ingress node, the service sub-layers of DetNet flows are aggregated into a service sub-layer of an aggregated DetNet flow
- Relay node aggregates the forwarding sub-layers DetNet flows into a forwarding sub-layer
- Relay node aggregates the service sub-layers of DetNet flows into a forwarding sub-layer
- Relay node aggregates the service sub-layers of DetNet flows into a service sub-layer of Aggregated DetNet flow
- Relay node aggregates the forwarding sub-layers of DetNet flow into a service sub-layer of Aggregated DetNet flow
- Transit node aggregates the forwarding sub-layers of DetNet flows into a forwarding sub-layer

Traffic requirements and traffic specification may be tracked for individual or aggregate flows but reserving resources and tracking the services in the aggregated flow is out of scope.

5. DetNet YANG Structure Considerations

The picture shows that the general structure of the DetNet YANG Model:



There are three instances in DetNet YANG Model: App-flow instance, service sub-layer instance and forwarding sub-layer instance, respectively corresponding to four parts of DetNet functions defined in section 3.

6. DetNet Configuration YANG Structures

```
module: ietf-detnet
  +--rw detnet
     +--rw traffic-profile* [profile-name]
     |  +--rw profile-name            string
     |  +--rw traffic-requirements
     |  |  +--rw min-bandwidth?                    uint64
     |  |  +--rw max-latency?                      uint32
     |  |  +--rw max-latency-variation?            uint32
     |  |  +--rw max-loss?                         uint32
     |  |  +--rw max-consecutive-loss-tolerance?   uint32
     |  |  +--rw max-misordering?                  uint32
     |  +--rw flow-spec
     |  |  +--rw interval?                uint32
     |  |  +--rw max-pkts-per-interval?   uint32
     |  |  +--rw max-payload-size?        uint32
     |  |  +--rw min-payload-size?        uint32
     |  |  +--rw min-pkts-per-interval?   uint32
     |  +--ro member-apps*            app-flow-ref
     |  +--ro member-services*        service-sub-layer-ref
     |  +--ro member-fwd-sublayers*   forwarding-sub-layer-ref
     +--rw app-flows
     |  +--rw app-flow* [name]
     |     +--rw name                          string
     |     +--rw app-flow-bidir-congruent?   boolean
     |     +--ro outgoing-service?           service-sub-layer-ref
     |     +--ro incoming-service?           service-sub-layer-ref
     |     +--rw traffic-profile?            traffic-profile-ref
     |     +--rw ingress
     |     |  +--rw name?                  string
     |     |  +--ro app-flow-status?       identityref
     |     |  +--rw interface?             if:interface-ref
     |     |  +--rw (data-flow-type)?
     |     |     +--:(tsn-app-flow)
     |     |     |  +--rw tsn-app-flow
     |     |     |     +--rw source-mac-address?
     |     |     |     |       yang:mac-address
     |     |     |     +--rw destination-mac-address?
     |     |     |     |       yang:mac-address
     |     |     |     +--rw ethertype?
     |     |     |     |       ethertypes:ethertype
     |     |     |     +--rw vlan-id?
     |     |     |     |       dot1q-types:vlanid
     |     |     |     +--rw pcp?
     |     |     |             dot1q-types:priority-type
     |     |     +--:(ip-app-flow)
     |     |     |  +--rw ip-app-flow
     |     |     |     +--rw src-ip-prefix?          inet:ip-prefix
     |     |     |     +--rw dest-ip-prefix?         inet:ip-prefix
     |     |     |     +--rw protocol-next-header?   uint8
     |     |     |     +--rw dscp?                   inet:dscp
     |     |     |     +--rw flow-label?
     |     |     |     |       inet:ipv6-flow-label
     |     |     |     +--rw source-port
     |     |     |     |  +--rw (port-range-or-operator)?
     |     |     |     |     +--:(range)
     |     |     |     |     |  +--rw lower-port    inet:port-number
     |     |     |     |     |  +--rw upper-port    inet:port-number
```

```
|      |      |      |     +--:(operator)
|      |      |      |        +--rw operator?     operator
|      |      |      |        +--rw port          inet:port-number
|      |      |      +--rw destination-port
|      |      |      |  +--rw (port-range-or-operator)?
|      |      |      |     +--:(range)
|      |      |      |     |  +--rw lower-port    inet:port-number
|      |      |      |     |  +--rw upper-port    inet:port-number
|      |      |      |     +--:(operator)
|      |      |      |        +--rw operator?     operator
|      |      |      |        +--rw port          inet:port-number
|      |      |      +--rw ipsec-spi?              ipsec-spi
|      |      +--:(mpls-app-flow)
|      |         +--rw mpls-app-flow
|      |            +--rw (label-space)?
|      |               +--:(context-label-space)
|      |               |  +--rw mpls-label-stack
|      |               |     +--rw entry* [id]
|      |               |        +--rw id               uint8
|      |               |        +--rw label?
|      |               |        |       rt-types:mpls-label
|      |               |        +--rw ttl?             uint8
|      |               |        +--rw traffic-class?   uint8
|      |               +--:(platform-label-space)
|      |                  +--rw label?
|      |                          rt-types:mpls-label
|      +--rw egress
|         +--rw name?             string
|         +--rw (application-type)?
|            +--:(ethernet)
|            |  +--rw ethernet
|            |     +--rw interface?   if:interface-ref
|            +--:(ip-mpls)
|               +--rw ip-mpls
|                  +--rw (next-hop-options)
|                     +--:(simple-next-hop)
|                     |  +--rw outgoing-interface?
|                     |  |       if:interface-ref
|                     |  +--rw (flow-type)?
|                     |     +--:(ip)
|                     |     |  +--rw next-hop-address?
|                     |     |          inet:ip-address
|                     |     +--:(mpls)
|                     |        +--rw mpls-label-stack
|                     |           +--rw entry* [id]
|                     |              +--rw id               uint8
|                     |              +--rw label?
|                     |              |       rt-types:mpls-label
|                     |              +--rw ttl?             uint8
|                     |              +--rw traffic-class?   uint8
|                     +--:(next-hop-list)
|                        +--rw next-hop* [hop-index]
|                           +--rw hop-index                uint8
|                           +--rw outgoing-interface?
|                           |       if:interface-ref
|                           +--rw (flow-type)?
|                              +--:(ip)
|                              |  +--rw next-hop-address?
```

```
      |                               |          inet:ip-address
      |                               +--:(mpls)
      |                                  +--rw mpls-label-stack
      |                                     +--rw entry* [id]
      |                                        +--rw id
      |                                        |       uint8
      |                                        +--rw label?
      |                                        |       rt-types:mpls-label
      |                                        +--rw ttl?
      |                                        |       uint8
      |                                        +--rw traffic-class?
      |                                                uint8
      +--rw service-sub-layer
      |  +--rw service-sub-layer-list* [name]
      |     +--rw name                        string
      |     +--rw service-rank?               uint8
      |     +--rw traffic-profile?            traffic-profile-ref
      |     +--rw service-protection
      |     |  +--rw service-protection-type?   service-protection-type
      |     |  +--rw sequence-number-length?    sequence-number-field
      |     +--rw service-operation-type?   service-operation-type
      |     +--rw incoming-type
      |     |  +--rw (incoming-type)
      |     |     +--:(app-flow)
      |     |     |  +--rw app-flow
      |     |     |     +--rw app-flow-list*   app-flow-ref
      |     |     +--:(service-aggregation)
      |     |     |  +--rw service-aggregation
      |     |     |     +--rw service-sub-layer*
      |     |     |              service-sub-layer-ref
      |     |     +--:(forwarding-aggregation)
      |     |     |  +--rw forwarding-aggregation
      |     |     |     +--rw forwarding-sub-layer*
      |     |     |              forwarding-sub-layer-ref
      |     |     +--:(service-id)
      |     |        +--rw service-id
      |     |           +--rw (detnet-flow-type)?
      |     |              +--:(ip-detnet-flow)
      |     |                 |  +--rw src-ip-prefix?
      |     |                 |  |       inet:ip-prefix
      |     |                 |  +--rw dest-ip-prefix?
      |     |                 |  |       inet:ip-prefix
      |     |                 |  +--rw protocol-next-header?     uint8
      |     |                 |  +--rw dscp?                     inet:dscp
      |     |                 |  +--rw flow-label?
      |     |                 |  |       inet:ipv6-flow-label
      |     |                 |  +--rw source-port
      |     |                 |  |  +--rw (port-range-or-operator)?
      |     |                 |  |     +--:(range)
      |     |                 |  |     |  +--rw lower-port
      |     |                 |  |     |  |       inet:port-number
      |     |                 |  |     |  +--rw upper-port
      |     |                 |  |     |          inet:port-number
      |     |                 |  |     +--:(operator)
      |     |                 |  |        +--rw operator?     operator
      |     |                 |  |        +--rw port
      |     |                 |  |                inet:port-number
      |     |                 |  +--rw destination-port
```

```
|      |              |  |  +--rw (port-range-or-operator)?
|      |              |  |     +--:(range)
|      |              |  |     |  +--rw lower-port
|      |              |  |     |  |       inet:port-number
|      |              |  |     |  +--rw upper-port
|      |              |  |     |          inet:port-number
|      |              |  |     +--:(operator)
|      |              |  |        +--rw operator?     operator
|      |              |  |        +--rw port
|      |              |  |                inet:port-number
|      |              |  +--rw ipsec-spi?                ipsec-spi
|      |              +--:(mpls-detnet-flow)
|      |                 +--rw (label-space)?
|      |                    +--:(context-label-space)
|      |                    |  +--rw mpls-label-stack
|      |                    |     +--rw entry* [id]
|      |                    |        +--rw id               uint8
|      |                    |        +--rw label?
|      |                    |        |       rt-types:mpls-label
|      |                    |        +--rw ttl?             uint8
|      |                    |        +--rw traffic-class?   uint8
|      |                    +--:(platform-label-space)
|      |                       +--rw label?
|      |                               rt-types:mpls-label
|      +--rw outgoing-type
|         +--rw (outgoing-type)
|            +--:(forwarding-sub-layer)
|            |  +--rw forwarding-sub-layer
|            |     +--rw service-outgoing-list*
|            |             [service-outgoing-index]
|            |        +--rw service-outgoing-index        uint8
|            |        +--rw (header-type)?
|            |        |  +--:(detnet-mpls-header)
|            |        |  |  +--rw mpls-label-stack
|            |        |  |     +--rw entry* [id]
|            |        |  |        +--rw id               uint8
|            |        |  |        +--rw label?
|            |        |  |        |       rt-types:mpls-label
|            |        |  |        +--rw ttl?             uint8
|            |        |  |        +--rw traffic-class?   uint8
|            |        |  +--:(detnet-ip-header)
|            |        |     +--rw src-ip-address?
|            |        |     |       inet:ip-address
|            |        |     +--rw dest-ip-address?
|            |        |     |       inet:ip-address
|            |        |     +--rw protocol-next-header?   uint8
|            |        |     +--rw dscp?
|            |        |     |       inet:dscp
|            |        |     +--rw flow-label?
|            |        |     |       inet:ipv6-flow-label
|            |        |     +--rw source-port?
|            |        |     |       inet:port-number
|            |        |     +--rw destination-port?
|            |        |             inet:port-number
|            |        +--rw forwarding-sub-layer*
|            |                forwarding-sub-layer-ref
|            +--:(service-sub-layer)
|            |  +--rw service-sub-layer
```

```
      |           |     +--rw aggregation-service-sub-layer?
      |           |     |       service-sub-layer-ref
      |           |     +--rw service-label
      |           |        +--rw mpls-label-stack
      |           |           +--rw entry* [id]
      |           |              +--rw id               uint8
      |           |              +--rw label?
      |           |              |       rt-types:mpls-label
      |           |              +--rw ttl?             uint8
      |           |              +--rw traffic-class?   uint8
      |           +--:(app-flow)
      |           |  +--rw app-flow
      |           |     +--rw app-flow-list*   app-flow-ref
      |           +--:(service-disaggregation)
      |           |  +--rw service-disaggregation
      |           |     +--rw service-sub-layer*
      |           |             service-sub-layer-ref
      |           +--:(forwarding-disaggregation)
      |              +--rw forwarding-disaggregation
      |                 +--rw forwarding-sub-layer*
      |                         forwarding-sub-layer-ref
      +--rw forwarding-sub-layer
         +--rw forwarding-sub-layer-list* [name]
            +--rw name                          string
            +--rw traffic-profile?              traffic-profile-ref
            +--rw forwarding-operation-type?
            |       forwarding-operations-type
            +--rw incoming-type
            |  +--rw (incoming-type)
            |     +--:(service-sub-layer)
            |     |  +--rw service-sub-layer
            |     |     +--rw service-sub-layer*
            |     |             service-sub-layer-ref
            |     +--:(forwarding-aggregation)
            |     |  +--rw forwarding-aggregation
            |     |     +--rw forwarding-sub-layer*
            |     |             forwarding-sub-layer-ref
            |     +--:(forwarding-id)
            |        +--rw forwarding-id
            |           +--rw interface?
            |           |       if:interface-ref
            |           +--rw (detnet-flow-type)?
            |              +--:(ip-detnet-flow)
            |              |  +--rw src-ip-prefix?
            |              |  |       inet:ip-prefix
            |              |  +--rw dest-ip-prefix?
            |              |  |       inet:ip-prefix
            |              |  +--rw protocol-next-header?     uint8
            |              |  +--rw dscp?                     inet:dscp
            |              |  +--rw flow-label?
            |              |  |       inet:ipv6-flow-label
            |              |  +--rw source-port
            |              |  |  +--rw (port-range-or-operator)?
            |              |  |     +--:(range)
            |              |  |     |  +--rw lower-port
            |              |  |     |  |       inet:port-number
            |              |  |     |  +--rw upper-port
            |              |  |     |          inet:port-number
```

```
|              |  |     +--:(operator)
|              |  |        +--rw operator?     operator
|              |  |        +--rw port
|              |  |                 inet:port-number
|              |  +--rw destination-port
|              |  |  +--rw (port-range-or-operator)?
|              |  |     +--:(range)
|              |  |     |  +--rw lower-port
|              |  |     |  |       inet:port-number
|              |  |     |  +--rw upper-port
|              |  |     |          inet:port-number
|              |  |     +--:(operator)
|              |  |        +--rw operator?     operator
|              |  |        +--rw port
|              |  |                 inet:port-number
|              |  +--rw ipsec-spi?                ipsec-spi
|              +--:(mpls-detnet-flow)
|                 +--rw (label-space)?
|                    +--:(context-label-space)
|                    |  +--rw mpls-label-stack
|                    |     +--rw entry* [id]
|                    |        +--rw id               uint8
|                    |        +--rw label?
|                    |        |       rt-types:mpls-label
|                    |        +--rw ttl?             uint8
|                    |        +--rw traffic-class?   uint8
|                    +--:(platform-label-space)
|                       +--rw label?
|                               rt-types:mpls-label
|
+--rw outgoing-type
   +--rw (outgoing-type)
      +--:(interface)
      |  +--rw interface
      |     +--rw (next-hop-options)
      |        +--:(simple-next-hop)
      |        |  +--rw outgoing-interface?
      |        |  |       if:interface-ref
      |        |  +--rw (flow-type)?
      |        |     +--:(ip)
      |        |     |  +--rw (operation-type)?
      |        |     |     +--:(ip-forwarding)
      |        |     |     |  +--rw next-hop-address?
      |        |     |     |          inet:ip-address
      |        |     |     +--:(mpls-over-ip-encapsulation)
      |        |     |        +--rw src-ip-address?
      |        |     |        |       inet:ip-address
      |        |     |        +--rw dest-ip-address?
      |        |     |        |       inet:ip-address
      |        |     |        +--rw protocol-next-header?
      |        |     |        |       uint8
      |        |     |        +--rw dscp?
      |        |     |        |       inet:dscp
      |        |     |        +--rw flow-label?
      |        |     |        |       inet:ipv6-flow-label
      |        |     |        +--rw source-port?
      |        |     |        |       inet:port-number
      |        |     |        +--rw destination-port?
      |        |     |                inet:port-number
```

```
|         |     +--:(mpls)
|         |        +--rw mpls-label-stack
|         |           +--rw entry* [id]
|         |              +--rw id               uint8
|         |              +--rw label?
|         |              |       rt-types:mpls-label
|         |              +--rw ttl?             uint8
|         |              +--rw traffic-class?   uint8
|         +--:(next-hop-list)
|            +--rw next-hop* [hop-index]
|               +--rw hop-index
|               |       uint8
|               +--rw outgoing-interface?
|               |       if:interface-ref
|               +--rw (flow-type)?
|                  +--:(ip)
|                  |  +--rw (operation-type)?
|                  |     +--:(ip-forwarding)
|                  |     |  +--rw next-hop-address?
|                  |     |          inet:ip-address
|                  |     +--:(mpls-over-ip-
|                  |           encapsulation)
|                  |        +--rw src-ip-address?
|                  |        |       inet:ip-address
|                  |        +--rw dest-ip-address?
|                  |        |       inet:ip-address
|                  |        +--rw protocol-next-header?
|                  |        |       uint8
|                  |        +--rw dscp?
|                  |        |       inet:dscp
|                  |        +--rw flow-label?
|                  |        |       inet:ipv6-flow-label
|                  |        +--rw source-port?
|                  |        |       inet:port-number
|                  |        +--rw destination-port?
|                  |                inet:port-number
|                  +--:(mpls)
|                     +--rw mpls-label-stack
|                        +--rw entry* [id]
|                           +--rw id
|                           |       uint8
|                           +--rw label?
|                           |       rt-types:mpls-label
|                           +--rw ttl?
|                           |       uint8
|                           +--rw traffic-class?
|                                   uint8
|
+--:(service-aggregation)
|  +--rw service-aggregation
|     +--rw aggregation-service-sub-layer?
|     |       service-sub-layer-ref
|     +--rw optional-forwarding-label
|        +--rw mpls-label-stack
|           +--rw entry* [id]
|              +--rw id               uint8
|              +--rw label?
|              |       rt-types:mpls-label
|              +--rw ttl?             uint8
```

```
                  |              +--rw traffic-class?   uint8
                  +--:(forwarding-sub-layer)
                  |  +--rw forwarding-sub-layer
                  |     +--rw aggregation-forwarding-sub-layer?
                  |     |       forwarding-sub-layer-ref
                  |     +--rw forwarding-label
                  |        +--rw mpls-label-stack
                  |           +--rw entry* [id]
                  |              +--rw id               uint8
                  |              +--rw label?
                  |              |       rt-types:mpls-label
                  |              +--rw ttl?             uint8
                  |              +--rw traffic-class?   uint8
                  +--:(service-sub-layer)
                  |  +--rw service-sub-layer
                  |     +--rw service-sub-layer*
                  |             service-sub-layer-ref
                  +--:(forwarding-disaggregation)
                     +--rw forwarding-disaggregation
                        +--rw forwarding-sub-layer*
                                forwarding-sub-layer-ref
```

7. DetNet Configuration YANG Model

```
<CODE BEGINS>
module ietf-detnet {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-detnet";
  prefix ietf-detnet;

  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6021 - Common YANG Data Types.";
  }
  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991 - Common YANG Data Types.";
  }
  import ietf-ethertypes {
    prefix ethertypes;
    reference
      "RFC 8519 - YANG Data Model for Network Access Control
                  Lists (ACLs).";
  }
  import ietf-routing-types {
    prefix rt-types;
    reference
      "RFC 8294 - Common YANG Data Types for the Routing Area.";
  }
  import ietf-packet-fields {
    prefix packet-fields;
    reference
      "RFC 8519 - YANG Data Model for Network Access Control Lists
       (ACLs).";
  }
  import ietf-interfaces {
    prefix if;
    reference
      "RFC 8343 - A YANG Data Model for Interface Management.";
  }
  import ieee802-dot1q-types {
    prefix dot1q-types;
    reference
      "IEEE 802.1Qcx-2020 - IEEE Standard for Local and Metropolitan
       Area Networks--Bridges and Bridged Networks Amendment 33: YANG
       Data Model for Connectivity Fault Management.";
  }

  organization
    "IETF DetNet Working Group";
  contact
    "WG Web:   <http://tools.ietf.org/wg/detnet/>
     WG List:  <mailto: detnet@ietf.org>

     Editor:   Xuesong Geng
               <mailto:gengxuesong@huawei.com>

     Editor:   Yeoncheol Ryoo
               <mailto:dbduscjf@etri.re.kr>
```

```
     Editor:   Don Fedyk
               <mailto:dfedyk@labn.net>;

     Editor:   Reshad Rahman
               <mailto:reshad@yahoo.com>

     Editor:   Mach Chen
               <mailto:mach.chen@huawei.com>

     Editor:   Zhenqiang Li
               <mailto:lizhenqiang@chinamobile.com>";
  description
    "This YANG module describes the parameters needed
     for DetNet flow configuration and flow status
     reporting.

     Copyright (c) 2021 IETF Trust and the persons identified as
     authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject to
     the license terms contained in, the Simplified BSD License set
     forth in Section 4.c of the IETF Trust's Legal Provisions
     Relating to IETF Documents
     (https://trustee.ietf.org/license-info).

     This version of this YANG module is part of RFC XXXX
     (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
     for full legal notices.

     The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL
     NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'NOT RECOMMENDED',
     'MAY', and 'OPTIONAL' in this document are to be interpreted as
     described in BCP 14 (RFC 2119) (RFC 8174) when, and only when,
     they appear in all capitals, as shown here. ";


  revision 2021-02-17 {
    description
      "initial revision";
    reference
      "RFC XXXX: draft-ietf-detnet-yang-10";
  }

  identity app-status {
    description
      "Base identity from which all application-status
       status types are derived.";
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  identity none {
    base app-status;
    description
      "This Application has no status. This type of status is
       expected when the configuration is incomplete.";
```

```
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  identity ready {
    base app-status;
    description
      "Application ingress/egress ready.";
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  identity failed {
    base app-status;
    description
      "Application ingres/egresss failed.";
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  identity out-of-service {
    base app-status;
    description
      "Application Administratively blocked.";
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  identity partial-failed {
    base app-status;
    description
      "This is an Application with one or more Egress ready, and one
       or more Egress failed.  The DetNet flow can be used if the
       Ingress is Ready.";
    reference
      "draft-ietf-detnet-flow-information-model Section 5.8";
  }

  typedef app-flow-ref {
    type leafref {
      path "/ietf-detnet:detnet"
         + "/ietf-detnet:app-flows"
         + "/ietf-detnet:app-flow"
         + "/ietf-detnet:name";
    }
    description
      "This is an Application Reference.";
  }

  typedef service-sub-layer-ref {
    type leafref {
      path "/ietf-detnet:detnet"
         + "/ietf-detnet:service-sub-layer"
         + "/ietf-detnet:service-sub-layer-list"
         + "/ietf-detnet:name";
    }
    description
      "This is a Service sub-layer Reference.";
```

```
  }

  typedef forwarding-sub-layer-ref {
    type leafref {
      path "/ietf-detnet:detnet"
         + "/ietf-detnet:forwarding-sub-layer"
         + "/ietf-detnet:forwarding-sub-layer-list"
         + "/ietf-detnet:name";
    }
    description
      "This is a Forwarding sub-layer Reference.";
  }

  typedef traffic-profile-ref {
    type leafref {
      path "/ietf-detnet:detnet"
         + "/ietf-detnet:traffic-profile"
         + "/ietf-detnet:profile-name";
    }
    description
      "This is a Traffic Profile Reference.";
  }

  typedef ipsec-spi {
    type uint32 {
      range "1..max";
    }
    description
      "IPsec Security Parameters Index.";
    reference
      "IETF RFC 6071";
  }

  typedef service-operation-type {
    type enumeration {
      enum service-initiation {
        description
          "This is an initiating service sub-layer encapsulation.";
      }
      enum service-termination {
        description
          "Operation for DetNet service sub-layer decapsulation.";
      }
      enum service-relay {
        description
          "Operation for DetNet service sub-layer swap.";
      }
      enum non-detnet {
        description
          "No operation for DetNet service sub-layer.";
      }
    }
    description
      "Operation type identifies the behavior for this service
       sub-layer instance. Operations are described as unidirectional
       but a service sub-layer may combine operation types.";
  }
```

```
  typedef forwarding-operations-type {
    type enumeration {
      enum impose-and-forward {
        description
          "This operation impose outgoing label(s) and forward to
           next-hop.";
        reference
          " A YANG Data Model for MPLS Base
            draft-ietf-mpls-base-yang.";
      }
      enum pop-and-forward {
        description
          "This operation pops the incoming label and forwards to
           the next-hop.";
        reference
          " A YANG Data Model for MPLS Base
            draft-ietf-mpls-base-yang.";
      }
      enum pop-impose-and-forward {
        description
          "This operation pops the incoming label, imposes one or
           more outgoing label(s) and forwards to the next-hop.";
        reference
          " A YANG Data Model for MPLS Base
            draft-ietf-mpls-base-yang.";
      }
      enum swap-and-forward {
        description
          "This operation swaps incoming label, with an outgoing
           label and forwards to the next-hop.";
        reference
          " A YANG Data Model for MPLS Base
            draft-ietf-mpls-base-yang.";
      }
      enum forward {
        description
          "This operation forward to next-hop.";
      }
      enum pop-and-lookup {
        description
          "This operation pops incoming label and performs a
           lookup.";
      }
    }
    description
      "MPLS operations types.  This is an enum modeled after the
       MPLS enum.  The first 4 enums are the same as A YANG Data
       Model for MPLS Base.  draft-ietf-mpls-base-yang.";
  }

  typedef service-protection-type {
    type enumeration {
      enum none {
        description
          "No service protection provided.";
      }
      enum replication {
        description
```

```
          "A Packet Replication Function (PRF) replicates DetNet
           flow packets and forwards them to one or more next hops in
           the DetNet domain.  The number of packet copies sent to
           each next hop is a DetNet flow specific parameter at the
           node doing the replication.  PRF can be implemented by an
           edge node, a relay node, or an end system.";
      }
      enum elimination {
        description
          "A Packet Elimination Function (PEF) eliminates duplicate
           copies of packets to prevent excess packets flooding the
           network or duplicate packets being sent out of the DetNet
           domain.  PEF can be implemented by an edge node, a relay
           node, or an end system.";
      }
      enum ordering {
        description
          "A Packet Ordering Function (POF) re-orders packets within
           a DetNet flow that are received out of order.  This
           function can be implemented by an edge node, a relay node,
           or an end system.";
      }
      enum elimination-ordering {
        description
          "A combination of PEF and POF that can be implemented by
           an edge node, a relay node, or an end system.";
      }
      enum elimination-replication {
        description
          "A combination of PEF and PRF that can be implemented by
           an edge node, a relay node, or an end system.";
      }
      enum elimination-ordering-replicaiton {
        description
          "A combination of PEF, POF and PRF that can be implemented
           by an edge node, a relay node, or an end system.";
      }
    }
    description
      "This typedef describes the service protection types.";
  }

  typedef sequence-number-generation-type {
    type enumeration {
      enum copy-from-app-flow {
        description
          "This type means copy the app-flow sequence number to the
           DetNet-flow.";
      }
      enum generate-by-detnet-flow {
        description
          "This type means generate the sequence number by the
           DetNet flow.";
      }
    }
    description
      "An enumeration for the sequence number behaviors supported.";
  }
```

```
typedef sequence-number-field {
  type enumeration {
    enum zero-sn {
      description
        "No DetNet sequence number field is used.";
    }
    enum short-sn {
      value 16;
      description
        "A 16-bit DetNet sequence number field is used.";
    }
    enum long-sn {
      value 28;
      description
        "A 28-bit DetNet sequence number field is used.";
    }
  }
  description
    "This type captures the sequence number behavior.";
}

grouping ip-header {
  description
    "This grouping captures the IPv4/IPv6 packet header
     information. it is modeled after existing fields.";
  leaf src-ip-address {
    type inet:ip-address-no-zone;
    description
      "The source IP address in the header.";
    reference
      "RFC 6021 Common YANG Data Types";
  }
  leaf dest-ip-address {
    type inet:ip-address-no-zone;
    description
      "The destination IP address in the header.";
    reference
      "RFC 6021 Common YANG Data Types";
  }
  leaf protocol-next-header {
    type uint8;
    description
      "Internet Protocol number.  Refers to the protocol of the
       payload.  In IPv6, this field is known as 'next-header',
       and if extension headers are present, the protocol is
       present in the 'upper-layer' header.";
    reference
      "RFC 791: Internet Protocol
       RFC 8200: Internet Protocol, Version 6 (IPv6)
       Specification.";
  }
  leaf dscp {
    type inet:dscp;
    description
      "The traffic class value in the header.";
    reference
      "RFC 6021 Common YANG Data Types";
```

```
    }
    leaf flow-label {
      type inet:ipv6-flow-label;
      description
        "The flow label value of the header.IPV6 only.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
    leaf source-port {
      type inet:port-number;
      description
        "The source port number.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
    leaf destination-port {
      type inet:port-number;
      description
        "The destination port number.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
  }

  grouping l2-header {
    description
      "The Ethernet or TSN packet header information.";
    leaf source-mac-address {
      type yang:mac-address;
      description
        "The source MAC address value of the Ethernet header.";
    }
    leaf destination-mac-address {
      type yang:mac-address;
      description
        "The destination MAC address value of the Ethernet header.";
    }
    leaf ethertype {
      type ethertypes:ethertype;
      description
        "The Ethernet packet type value of the Ethernet header.";
    }
    leaf vlan-id {
      type dot1q-types:vlanid;
      description
        "The VLAN value of the Ethernet header.";
      reference
        "IEEE 802.1Qcx-2020.";
    }
    leaf pcp {
      type dot1q-types:priority-type;
      description
        "The priority value of the Ethernet header.";
      reference
        "IEEE 802.1Qcx-2020.";
    }
  }
```

```
  grouping destination-ip-port-id {
    description
      "The TCP/UDP port(source/destination) identification
       information.";
    container destination-port {
      uses packet-fields:port-range-or-operator;
      description
         "This grouping captures the destination port fields.";
    }
  }

  grouping source-ip-port-id {
    description
      "The TCP/UDP port(source/destination) identification
       information.";
    container source-port {
      uses packet-fields:port-range-or-operator;
      description
         "This grouping captures the source port fields.";
    }
  }

  grouping ip-flow-id {
    description
      "The IPv4/IPv6 packet header identification information.";
    leaf src-ip-prefix {
      type inet:ip-prefix;
      description
        "The source IP prefix.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
    leaf dest-ip-prefix {
      type inet:ip-prefix;
      description
        "The destination IP prefix.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
    leaf protocol-next-header {
      type uint8;
      description
        "Internet Protocol number.  Refers to the protocol of the
         payload.  In IPv6, this field is known as 'next-header', and
         if extension headers are present, the protocol is present in
         the 'upper-layer' header.";
      reference
        "RFC 791: Internet Protocol
          RFC 8200: Internet Protocol, Version 6 (IPv6)
         Specification.";
    }
    leaf dscp {
      type inet:dscp;
      description
        "The traffic class value in the header.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
```

```
    leaf flow-label {
      type inet:ipv6-flow-label;
      description
        "The flow label value of the header.";
      reference
        "RFC 6021 Common YANG Data Types";
    }
    uses source-ip-port-id;
    uses destination-ip-port-id;
    leaf ipsec-spi {
      type ipsec-spi;
      description
      "IPsec Security Parameters Index of the Security
       Association.";
      reference
        "IETF RFC 6071 IP Security (IPsec) and Internet Key Exchange
         (IKE) Document Roadmap.";
    }
  }

  grouping mpls-flow-id {
    description
      "The MPLS packet header identification information.";
    choice label-space {
      description
        "Designates the label space being used.";
      case context-label-space {
        uses rt-types:mpls-label-stack;
      }
      case platform-label-space {
        leaf label {
          type rt-types:mpls-label;
          description
           "This is the case for Platform label space.";
        }
      }
    }
  }

  grouping data-flow-spec {
    description
      "app-flow identification.";
    choice data-flow-type {
      description
        "The Application flow type choices.";
      container tsn-app-flow {
        uses l2-header;
        description
          "The L2 header for application.";
      }
      container ip-app-flow {
        uses ip-flow-id;
        description
          "The IP header for application.";
      }
      container mpls-app-flow {
        uses mpls-flow-id;
        description
```

```
          "The MPLS header for application.";
      }
    }
  }

  grouping detnet-flow-spec {
    description
      "detnet-flow identification.";
    choice detnet-flow-type {
      description
        "The Detnet flow type choices.";
      case ip-detnet-flow {
        uses ip-flow-id;
      }
      case mpls-detnet-flow {
        uses mpls-flow-id;
      }
    }
  }

  grouping app-flows-group {
    description
      "Incoming or outgoing app-flow reference group.";
    leaf-list app-flow-list {
      type app-flow-ref;
      description
        "List of ingress or egress app-flows.";
    }
  }

  grouping service-sub-layer-group {
    description
      "Incoming or outgoing service sub-layer reference group.";
    leaf-list service-sub-layer {
      type service-sub-layer-ref;
      description
        "List of incoming or outgoing service sub-layers that have
         to aggregate or disaggregate.";
    }
  }

  grouping forwarding-sub-layer-group {
    description
      "Incoming or outgoing forwarding sub-layer reference group.";
    leaf-list forwarding-sub-layer {
      type forwarding-sub-layer-ref;
      description
        "List of incoming or outgoing forwarding sub-layers that
         have to aggregate or disaggregate.";
    }
  }

  grouping detnet-header {
    description
      "DetNet header info for DetNet encapsulation or swap.";
    choice header-type {
      description
      "The choice of DetNet header type.";
```

```
      case detnet-mpls-header {
        description
          "MPLS label stack for DetNet MPLS encapsulation or
           forwarding.";
        uses rt-types:mpls-label-stack;
      }
      case detnet-ip-header {
        description
          "IPv4/IPv6 packet header for DetNet IP encapsulation.";
        uses ip-header;
      }
    }
  }

  grouping detnet-app-next-hop-content {
    description
      "Generic parameters of DetNet next hops.";
    choice next-hop-options {
      mandatory true;
      description
        "Options for next hops.  It is expected that further cases
         will be added through
         augments from other modules, e.g., for recursive
         next hops.";
      case simple-next-hop {
        description
          "This case represents a simple next hop consisting of the
           next-hop address and/or outgoing interface.
           Modules for address families MUST augment this case with a
           leaf containing a next-hop address of that address
           family.";
        leaf outgoing-interface {
          type if:interface-ref;
          description
            "The outgoing interface, if this is a whole interface.";
        }
        choice flow-type {
          description
            "The flow type choices.";
          case ip {
            leaf next-hop-address {
              type inet:ip-address-no-zone;
              description
               "The IP next hop case.";
            }
          }
          case mpls {
            uses rt-types:mpls-label-stack;
            description
              "The MPLS Label stack next hop case.";
          }
        }
      }
      case next-hop-list {
        description
          "Container for multiple next hops.";
        list next-hop {
          key "hop-index";
```

```
          description
            "An entry in a next-hop list.  Modules for address
             families MUST augment this list with a leaf containing a
             next-hop address of that address family.";
          leaf hop-index {
            type uint8;
            description
              "A user-specified identifier utilized to uniquely
               reference the next-hop entry in the next-hop list.
               The value of this index has no semantic meaning other
               than for referencing the entry.";
          }
          leaf outgoing-interface {
            type if:interface-ref;
            description
              "Name of the outgoing interface.";
          }
          choice flow-type {
            description
              "The flow types supported.";
            case ip {
              leaf next-hop-address {
                type inet:ip-address-no-zone;
                description
                  "This is the IP flow type next hop.";
              }
            }
            case mpls {
              uses rt-types:mpls-label-stack;
            }
          }
        }
      }
    }
  }

  grouping detnet-forwarding-next-hop-content {
    description
      "Generic parameters of DetNet next hops.";
    choice next-hop-options {
      mandatory true;
      description
        "Options for next hops.
         It is expected that further cases will be added through
         augments from other modules, e.g., for recursive
         next hops.";
       case simple-next-hop {
        description
          "This case represents a simple next hop consisting of the
           next-hop address and/or outgoing interface.
           Modules for address families MUST augment this case with a
           leaf containing a next-hop address of that address
           family.";
        leaf outgoing-interface {
          type if:interface-ref;
          description
            "This is the interface as an outgoing type.";
        }
```

```
          choice flow-type {
            description
              "These are the flow type next hop choices.";
            case ip {
              choice operation-type {
                description
                  "This is the IP forwarding operation choices.";
                case ip-forwarding {
                  leaf next-hop-address {
                    type inet:ip-address-no-zone;
                    description
                      "This is an IP address as a next hop.";
                  }
                }
                case mpls-over-ip-encapsulation {
                  uses ip-header;
                }
              }
            }
            case mpls {
              uses rt-types:mpls-label-stack;
            }
          }
        }
        case next-hop-list {
          description
            "Container for multiple next hops.";
          list next-hop {
            key "hop-index";
            description
              "An entry in a next-hop list.  Modules for address
               families MUST augment this list with a leaf containing a
               next-hop address of that address family.";
            leaf hop-index {
              type uint8;
              description
                "The value of the index for a hop.";
            }
            leaf outgoing-interface {
              type if:interface-ref;
              description
                "This is a whole interface as the next hop.";
            }
            choice flow-type {
              description
                "These are the flow type next hop choices.";
              case ip {
                choice operation-type {
                  description
                    "These are the next hop choices.";
                  case ip-forwarding {
                    leaf next-hop-address {
                      type inet:ip-address-no-zone;
                      description
                        "This is an IP address as a next hop.";
                    }
                  }
                  case mpls-over-ip-encapsulation {
```

```
                  uses ip-header;
                }
              }
            }
            case mpls {
              uses rt-types:mpls-label-stack;
            }
          }
        }
      }
    }
  }

  container detnet {
    description
      "The top level DetNet container. This contains
       applications, service sub-layers and forwarding sub-layers
       as well as the traffic profiles.";
    list traffic-profile {
      key "profile-name";
      description
        "A traffic profile.";
      leaf profile-name {
        type string;
        description
          "An Aggregation group ID. Zero means the service is not
           part of a group.";
      }
      container traffic-requirements {
        description
          "This defines the attributes of the App-flow
           regarding bandwidth, latency, latency variation, loss, and
           misordering tolerance.";
        reference
          "draft-ietf-detnet-flow-information-model Section 4.2";
        leaf min-bandwidth {
          type uint64;
          units "bps";
          description
            "This is the minimum bandwidth that has to be
             guaranteed for the DetNet service.  MinBandwidth is
             specified in octets per second.";
        }
        leaf max-latency {
          type uint32;
          units "nanoseconds";
          description
            "This is the maximum latency from Ingress to
             Egress(es) for a single packet of the DetNet flow.
             MaxLatency is specified as an integer number of
             nanoseconds.";
        }
        leaf max-latency-variation {
          type uint32;
          units "nanoseconds";
          description
            "This is the difference between the
             minimum and the maximum end-to-end one-way latency.
```

```
            MaxLatencyVariation is specified as an integer number of
            nanoseconds.";
        }
        leaf max-loss {
          type uint32;
          description
            "This defines the maximum Packet Loss Ratio (PLR)
            parameter for the DetNet service between the Ingress and
            Egress(es) of the DetNet domain.";
        }
        leaf max-consecutive-loss-tolerance {
          type uint32;
          units "packets";
          description
            "Some applications have special loss requirement, such
            as MaxConsecutiveLossTolerance.  The maximum consecutive
            loss tolerance parameter describes the maximum number of
            consecutive packets whose loss can be tolerated.  The
            maximum consecutive loss tolerance can be measured for
            example based on sequence number.";
        }
        leaf max-misordering {
          type uint32;
          units "packets";
          description
            "This describes the tolerable maximum number
            of packets that can be received out of order.  The
            maximum allowed misordering can be measured for example
            based on sequence number.  The value zero for the
            maximum allowed misordering indicates that in order
            delivery is required, misordering cannot be tolerated.";
        }
      }
      container flow-spec {
        description
          "Flow-specification specifies how the Source transmits
           packets for the flow.  This is the promise/request of the
           Source to the network.  The network uses this flow
           specification to allocate resources and adjust queue
           parameters in network nodes.";
        reference
          "draft-ietf-detnet-flow-information-model Section 5.5";
        leaf interval {
          type uint32;
          units "nanoseconds";
          description
            "The period of time in which the traffic
             specification cannot be exceeded.";
        }
        leaf max-pkts-per-interval {
          type uint32;
          description
            "The maximum number of packets that the
             source will transmit in one interval.";
        }
        leaf max-payload-size {
          type uint32;
          description
```

```
            "The maximum payload size that the source
             will transmit.";
        }
        leaf min-payload-size {
          type uint32;
          description
            "The minimum payload size that the source
             will transmit.";
        }
        leaf min-pkts-per-interval {
          type uint32;
          description
            "The minimum number of packets that the
             source will transmit in one interval.";
        }
      }
      leaf-list member-apps {
        type app-flow-ref;
        config false;
        description
          "Applications attached to this profile.";
      }
      leaf-list member-services {
        type service-sub-layer-ref;
        config false;
        description
          "Services attached to this profile.";
      }
      leaf-list member-fwd-sublayers {
        type forwarding-sub-layer-ref;
        config false;
        description
          "Forwarding sub-layer attached to this profile.";
      }
    }
    container app-flows {
      description
        "The DetNet app-flow configuration.";
      reference
        "draft-ietf-detnet-flow-information-model Section 4.1";
      list app-flow {
        key "name";
        description
          "A unique (management) identifier of the App-flow.";
        leaf name {
          type string;
          description
            "A unique (management) identifier of the App-flow.";
          reference
            "draft-ietf-detnet-flow-information-model
             Sections 4.1, 5.1";
        }
        leaf app-flow-bidir-congruent {
          type boolean;
          default false;
          description
            "Defines the data path requirement of the App-flow
             whether it must share the same data path and physical
```

```
             path for both directions through the network, e.g., to
             provide congruent paths in the two directions.";
        reference
          "draft-ietf-detnet-flow-information-model
           Section 4.2";
      }
      leaf outgoing-service {
        type service-sub-layer-ref;
        config false;
        description
          "Binding to this applications outgoing
           service.";
      }
      leaf incoming-service {
        type service-sub-layer-ref;
        config false;
        description
          "Binding to this applications incoming service.";
      }
      leaf traffic-profile {
        type traffic-profile-ref;
        description
          "The Traffic Profile for this group.";
      }
      container ingress {
        description
          "Ingress DetNet application flows or a compound flow.";
        leaf name {
          type string;
          description
            "Ingress DetNet application.";
        }
        leaf app-flow-status {
          type identityref {
            base app-status;
          }
          config false;
          description
            "Status of ingress application flow.";
          reference
            "draft-ietf-detnet-flow-information-model Sections
             4.1, 5.8";
        }
        leaf interface {
          type if:interface-ref;
          description
          "Interface is used for any service type where a whole
          interface is mapped to the applications. It may be
          further filtered by type.";
        }
        uses data-flow-spec;
      } //End of app-ingress
      container egress {
        description
          "Route's next-hop attribute.";
        leaf name {
          type string;
          description
```

```
              "Egress DetNet application.";
          }
          choice application-type {
            description
              "This is the application type choices.";
            container ethernet {
              description
                "This is TSN unaware traffic that maps to an
                interface.";
              leaf interface {
                type if:interface-ref;
                description
                  "This is an Ethernet or TSN interfaces.";
              }
            }
            container ip-mpls {
              description
                "This is IP or MPLS DetNet application types.";
              uses detnet-app-next-hop-content;
            }
          }
        }
      }
    }
    container service-sub-layer {
      description
        "The DetNet service sub-layer configuration.";
      list service-sub-layer-list {
        key "name";
        description
          "Services are indexed by name.";
        leaf name {
          type string;
          description
            "The name of the DetNet service sub-layer.";
        }
        leaf service-rank {
          type uint8;
          description
            "The DetNet rank for this service.";
          reference
            "draft-ietf-detnet-flow-information-model Section 5.7.";
        }
        leaf traffic-profile {
          type traffic-profile-ref;
          description
            "The Traffic Profile for this service.";
        }
        container service-protection {
          description
            "This is the service protection type an sequence number
             options.";
          leaf service-protection-type {
            type service-protection-type;
            description
              "The DetNet service protection type such as PRF, PEF,
               PEOF,PERF, and PEORF.";
            reference
```

```
            "draft-ietf-detnet-data-plane-framework Section 4.3";
        }
        leaf sequence-number-length {
          type sequence-number-field;
          description
            "Sequence number field length can be one of 0 (none),
             16-bits or 28-bits.";
        }
      }
      leaf service-operation-type {
        type service-operation-type;
        description
          "This is the service operation type for this service
           sub-layer;";
      }
      container incoming-type {
        description
          "The DetNet service sub-layer incoming configuration.";
        choice incoming-type {
          mandatory true;
          description
            "A service sub-layer may have App flows or other
             service sub-layers.";
          container app-flow {
            description
              "This service sub-layer is related to the app-flows
               of the upper layer and provide ingress proxy or
               ingress aggregation at the ingress node.";
            uses app-flows-group;
          }
          container service-aggregation {
            description
              "This service sub-layer is related to the service
               sub-layer of the upper layer and provide
               service-to-service aggregation at the ingress node
               or relay node.";
            uses service-sub-layer-group;
          }
          container forwarding-aggregation {
            description
              "This service sub-layer is related to the forwarding
               sub-layer of the upper layer and provide
               forwarding-to-service aggregation at the ingress
               node or relay node.";
            uses forwarding-sub-layer-group;
          }
          container service-id {
            description
              "This service sub-layer is related to the service or
               forwarding sub-layer of the lower layer and provide
               DetNet service relay or termination at the relay
               node or egress node.";
            uses detnet-flow-spec;
          }
        }
      }
      container outgoing-type {
        description
```

```
            "The DetNet service sub-layer outgoing configuration.";
          choice outgoing-type {
            mandatory true;
            description
              "The out-going type may be a forwarding Sub-layer or a
               service sub-layer or ? types need to be named.";
            container forwarding-sub-layer {
              description
                "This service sub-layer is sent to the forwarding
                 sub-layers of the lower layer for DetNet service
                 forwarding or service-to-forwarding aggregation at
                 the ingress node or relay node.  When the operation
                 type is service-initiation, The service sub-layer
                 encapsulates the DetNet Control-Word and services
                 label, which are for individual DetNet flow when the
                 incoming type is app-flow and for aggregated DetNet
                 flow when the incoming type is service or
                 forwarding.  The service sub-layer swaps the service
                 label when the operation type is service-relay.";
              list service-outgoing-list {
                key "service-outgoing-index";
                description
                  "List of the outgoing service
                   that separately for each node
                   where services will be eliminated.";
                leaf service-outgoing-index {
                  type uint8;
                  description
                    "This index allows a list of multiple outgoing
                     forwarding sub-layers";
                }
                uses detnet-header;
                uses forwarding-sub-layer-group;
              }
            }
            container service-sub-layer {
              description
                "This service sub-layer is sent to the service
                 sub-layers of the lower layer for service-to-service
                 aggregation at the ingress node or relay node.  The
                 service sub-layer encapsulates the DetNet
                 Control-Word and S-label when the operation type is
                 service-initiation, and swaps the S-label when the
                 operation type is service-relay.";
              leaf aggregation-service-sub-layer {
                type service-sub-layer-ref;
                description
                  "reference point of the service-sub-layer
                   at which this service will be aggregated.";
              }
              container service-label {
                description
                  "This is the MPLS service sub-layer label.";
                uses rt-types:mpls-label-stack;
              }
            }
            container app-flow {
              description
```

```
                "This service sub-layer is sent to the app-flow of
                 the upper layer for egress proxy at the egress node,
                 and decapsulates the DetNet Control-Word and S-label
                 for individual DetNet service.  This outgoing type
                 only can be chosen when the operation type is
                 service-termination.";
              uses app-flows-group;
            }
            container service-disaggregation {
              description
                "This service sub-layer is sent to the service
                 sub-layer of the upper layer for service-to-service
                 disaggregation at the relay node or egress node, and
                 decapsulates the DetNet Control-Word and A-label for
                 aggregated DetNet service.  This outgoing type only
                 can be chosen when the operation type is
                 service-termination.";
              uses service-sub-layer-group;
            }
            container forwarding-disaggregation {
              description
                "This service sub-layer is sent to the forwarding
                 sub-layer of the upper layer for
                 forwarding-to-service disaggregation at the relay
                 node or egress node, and decapsulates the DetNet
                 Control-Word and A-label for aggregated DetNet
                 service.  This outgoing type only can be chosen when
                 the operation type is service-termination.";
              uses forwarding-sub-layer-group;
            }
          }
        }
      }
    }
    container forwarding-sub-layer {
      description
        "The DetNet forwarding sub-layer configuration.";
      list forwarding-sub-layer-list {
        key "name";
        description
          "The List is one or more DetNet Traffic types.";
        leaf name {
          type string;
          description
            "The name of the DetNet forwarding sub-layer.";
        }
        leaf traffic-profile {
          type traffic-profile-ref;
          description
            "The Traffic Profile for this group.";
        }
        leaf forwarding-operation-type {
          type forwarding-operations-type;
          description
            "This is the forwarding operation types
             impose-and-forward, pop-and-forward,
             pop-impose-and-forward, forward, pop-and-lookup.";
        }
```

```
          container incoming-type {
            description
              "The DetNet forwarding sub-layer incoming
               configuration.";
            choice incoming-type {
              mandatory true;
              description
                "Cases of incoming types.";
              container service-sub-layer {
                description
                  "This forwarding sub-layer is related to the service
                   sub-layers of the upper layer and provide DetNet
                   forwarding or service-to-forwarding aggregation at
                   the ingress node or relay node.";
                uses service-sub-layer-group;
              }
              container forwarding-aggregation {
                description
                  "This forwarding sub-layer is related to the
                   forwarding sub-layer of the upper layer and provide
                   forwarding-to-forwarding aggregation at the ingress
                   node or relay node or transit node.";
                uses forwarding-sub-layer-group;
              }
              container forwarding-id  {
                description
                  "This forwarding sub-layer is related to all of the
                   lower layer and provide DetNet forwarding swap or
                   termination at the transit node or relay node or
                   egress node.";
                leaf interface {
                  type if:interface-ref;
                  description
                    "This is the interface associated with the
                     forwarding sub-layer.";
                }
                uses detnet-flow-spec;
              }
            }
          }
          container outgoing-type {
            description
              "The DetNet forwarding sub-layer outbound
               configuration.";
            choice outgoing-type {
              mandatory true;
              description
                "This is when a service connected directly to an
                interface with no forwarding sub-layer.";
              container
                interface {
                description
                  "This forwarding sub-layer is sent to the interface
                   for send to next-hop at the ingress node or relay
                   node or transit node.";
                uses detnet-forwarding-next-hop-content;
              }
              container service-aggregation {
```

```
            description
              "This forwarding sub-layer is sent to the service
               sub-layers of the lower layer for
               forwarding-to-service aggregation at the ingress
               node or relay node.";
            leaf aggregation-service-sub-layer {
              type service-sub-layer-ref;
              description
                "This is reference to the service sub-layer.";
            }
            container optional-forwarding-label {
              description
                "This is the optional forwarding label for service
                 aggregation.";
              uses rt-types:mpls-label-stack;
            }
          }
          container forwarding-sub-layer {
            description
              "This forwarding sub-layer is sent to the forwarding
               sub-layers of the lower layer for
               forwarding-to-forwarding aggregation at the ingress
               node or relay node or transit node.";
            leaf aggregation-forwarding-sub-layer {
              type forwarding-sub-layer-ref;
              description
                "This is reference to the forwarding sub-layer.";
            }
            container forwarding-label {
              description
                "This is the forwarding label for forwarding
                 sub-layer aggregation.";
              uses rt-types:mpls-label-stack;
            }
          }
          container service-sub-layer {
            description
              "This forwarding sub-layer is sent to the service
               sub-layer of the upper layer and decapsulate the
               F-label for DetNet service or service-to-forwarding
               disaggregation at the relay node or egress node.
               This outgoing type only can be chosen when the
               operation type is pop-and-lookup.";
            uses service-sub-layer-group;
          }
          container forwarding-disaggregation {
            description
              "This forwarding sub-layer is sent to the forwarding
               sub-layer of the upper layer and decapsulate the
               F-label for forwarding-to-forwarding disaggregation
               at the transit node or relay node or egress node.
               This outgoing type only can be chosen when the
               operation type is pop-and-lookup.";
            uses forwarding-sub-layer-group;
          }
        }
      }
    }
```

```
    }
  }
}

<CODE ENDS>
```

8. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

9. Security Considerations

<TBD>

10. Acknowledgements

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <https://www.rfc-editor.org/info/rfc8655>.

11.2. Informative References

[I-D.ietf-detnet-flow-information-model] Varga, B., Farkas, J., Cummings, R., Jiang, Y., and D. Fedyk, "DetNet Flow and Service Information Model", Work in Progress, Internet-Draft, draft-ietf-detnet-flow-information-model-14, 24 January 2021, <http://www.ietf.org/internet-drafts/draft-ietf-detnet-flow-information-model-14.txt>.

Appendix A. Examples

The following examples are provided. These examples are tested with Yanglint and use operational output to exercise both config true and config false objects.

The following are examples of aggregation and disaggregation at various points in Detnet. Figures are provided in the PDF version of this document.

A.1. Example A-1 JSON Configuration/Operational

This illustrates simple aggregation. Ingress node 1 aggregates App flows 0 and 1 into a service sub-layer of DetNet flow 1. Two ways of illustrating this follow, then the JSON operational data model corresponding to the diagrams follows.

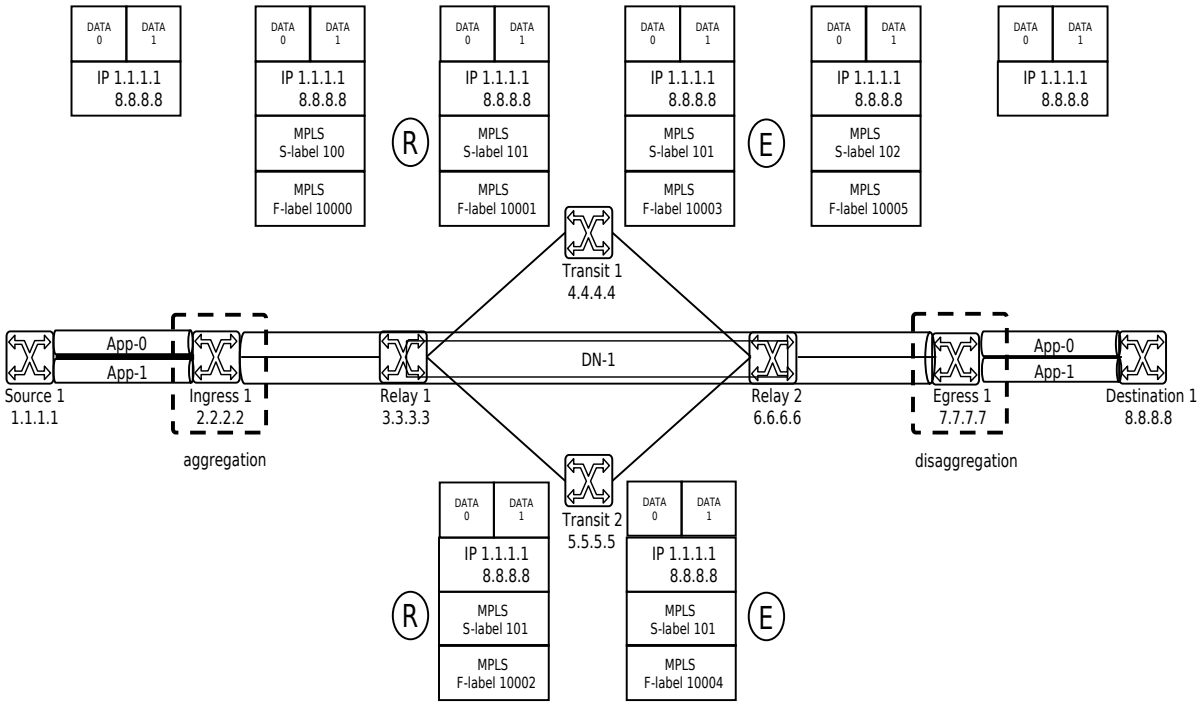


Figure 1: Case A-1 Example JSON Operational/Configuration

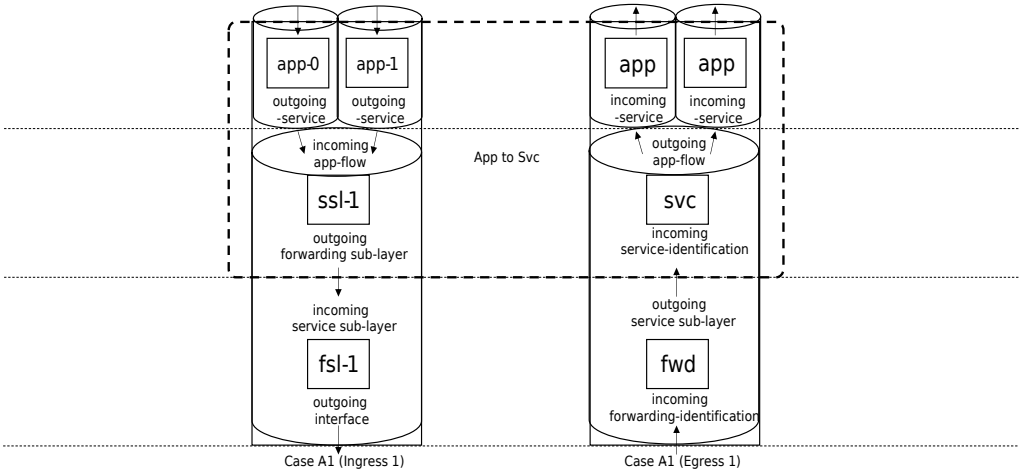


Figure 2: Case A-1 Example JSON Operational/Configuration

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "app-flows": {
      "app-flow": [
        {
          "name": "app-0",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-1",
          "traffic-profile": "pf-1",
          "ingress": {
            "app-flow-status": "ready",
            "interface": "eth0",
            "ip-app-flow": {
```

```
              "src-ip-prefix": "1.1.1.1/32",
              "dest-ip-prefix": "8.8.8.8/32",
              "dscp": 6
            }
          }
        },
        {
          "name": "app-1",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-1",
          "traffic-profile": "pf-1",
          "ingress": {
            "app-flow-status": "ready",
            "interface": "eth0",
            "ip-app-flow": {
              "src-ip-prefix": "1.1.1.2/32",
              "dest-ip-prefix": "8.8.8.9/32",
              "dscp": 7
            }
          }
        }
      ]
    },
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 200000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
        "flow-spec": {
          "interval": 5,
          "max-pkts-per-interval": 10,
          "max-payload-size": 1500,
          "min-payload-size": 100,
          "min-pkts-per-interval": 1
        },
        "member-apps": [
          "app-0",
          "app-1"
        ]
      },
      {
        "profile-name": "pf-2",
        "traffic-requirements": {
          "min-bandwidth": "200000000",
          "max-latency": 100000000,
          "max-latency-variation": 200000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
        "flow-spec": {
          "interval": 5,
```

```
          "max-pkts-per-interval": 10,
          "max-payload-size": 1500,
          "min-payload-size": 100,
          "min-pkts-per-interval": 1
        },
        "member-services": [
          "ssl-1"
        ]
      },
      {
        "profile-name": "pf-3",
        "flow-spec": {
          "interval": 5,
          "max-pkts-per-interval": 10,
          "max-payload-size": 1500
        },
        "member-fwd-sublayers": [
          "fsl-1"
        ]
      }
    ],
    "service-sub-layer": {
      "service-sub-layer-list": [
        {
          "name": "ssl-1",
          "service-rank": 10,
          "traffic-profile": "pf-2",
          "service-operation-type": "service-initiation",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "incoming-type": {
            "app-flow": {
              "app-flow-list": [
                "app-0",
                "app-1"
              ]
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 100
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-1"
                  ]
                }
              ]
```

```
            }
          }
        }
      ]
    },
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10000
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 3: Example A-1 DetNet JSON configuration

A.2. Example B-1 XML Config: Aggregation using a Forwarding Sub-layer

This illustrates aggrgation in the service sub-layers of DetNet. Flows 1 and 2 are aggregated into a forwarding sub-layer. A diagram illustrating this case is shown and then the corresponding XML operational data follows.

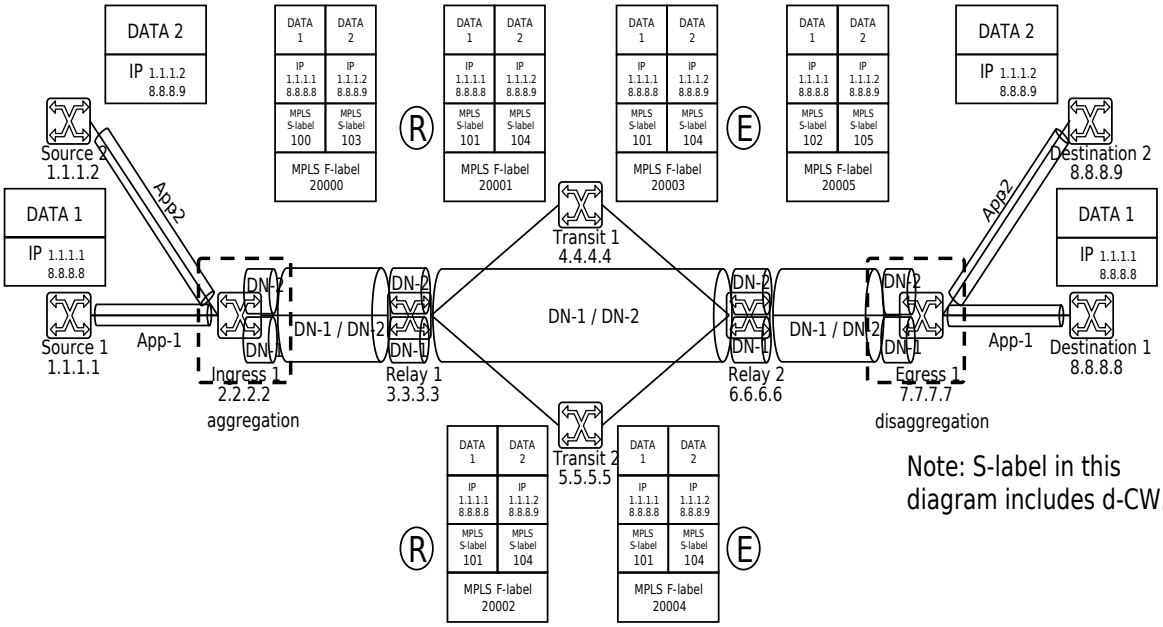


Figure 4: Case B-1 Example XML Config: Aggregation using a Forwarding Sub-layer

```
<interfaces
  xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"
  xmlns:ia="urn:ietf:params:xml:ns:yang:iana-if-type">
    <interface>
      <name>eth0</name>
      <type>ia:ethernetCsmacd</type>
      <oper-status>up</oper-status>
      <statistics>
         <discontinuity-time>2020-12-18T23:59:00Z</discontinuity-time>
      </statistics>
    </interface>
    <interface>
      <name>eth1</name>
      <type>ia:ethernetCsmacd</type>
      <oper-status>up</oper-status>
      <statistics>
         <discontinuity-time>2020-12-18T23:59:00Z</discontinuity-time>
      </statistics>
    </interface>
    <interface>
      <name>eth2</name>
      <type>ia:ethernetCsmacd</type>
      <oper-status>up</oper-status>
      <statistics>
         <discontinuity-time>2020-12-18T23:59:00Z</discontinuity-time>
      </statistics>
    </interface>
    <interface>
      <name>eth3</name>
      <type>ia:ethernetCsmacd</type>
      <oper-status>up</oper-status>
      <statistics>
         <discontinuity-time>2020-12-18T23:59:00Z</discontinuity-time>
      </statistics>
    </interface>
    <interface>
      <name>eth4</name>
      <type>ia:ethernetCsmacd</type>
      <oper-status>up</oper-status>
      <statistics>
         <discontinuity-time>2020-12-18T23:59:00Z</discontinuity-time>
      </statistics>
    </interface>
  </interfaces>
<detnet
  xmlns="urn:ietf:params:xml:ns:yang:ietf-detnet">
  <app-flows>
    <app-flow>
      <name>app-1</name>
      <app-flow-bidir-congruent>false</app-flow-bidir-congruent>
      <outgoing-service>ssl-1</outgoing-service>
       <traffic-profile>1</traffic-profile>
      <ingress>
        <app-flow-status>ready</app-flow-status>
        <interface>eth0</interface>
        <ip-app-flow>
          <src-ip-prefix>1.1.1.1/32</src-ip-prefix>
```

```
          <dest-ip-prefix>8.8.8.8/32</dest-ip-prefix>
          <dscp>6</dscp>
        </ip-app-flow>
      </ingress>
    </app-flow>
    <app-flow>
      <name>app-2</name>
      <app-flow-bidir-congruent>false</app-flow-bidir-congruent>
      <outgoing-service>ssl-2</outgoing-service>
       <traffic-profile>1</traffic-profile>
      <ingress>
        <app-flow-status>ready</app-flow-status>
        <interface>eth1</interface>
        <ip-app-flow>
          <src-ip-prefix>1.1.1.2/32</src-ip-prefix>
          <dest-ip-prefix>8.8.8.9/32</dest-ip-prefix>
          <dscp>7</dscp>
        </ip-app-flow>
        <dscp>7</dscp>
      </ingress>
    </app-flow>
  </app-flows>
  <traffic-profile>
    <profile-name>1</profile-name>
    <traffic-requirements>
      <min-bandwidth>100000000</min-bandwidth>
      <max-latency>100000000</max-latency>
      <max-latency-variation>200000000</max-latency-variation>
      <max-loss>2</max-loss>
      <max-consecutive-loss-tolerance>5</max-consecutive-loss-tolerance>
      <max-misordering>0</max-misordering>
    </traffic-requirements>
    <member-apps>app-1</member-apps>
    <member-apps>app-2</member-apps>
  </traffic-profile>
  <traffic-profile>
    <profile-name>2</profile-name>
    <traffic-requirements>
      <min-bandwidth>100000000</min-bandwidth>
      <max-latency>100000000</max-latency>
      <max-latency-variation>200000000</max-latency-variation>
      <max-loss>2</max-loss>
      <max-consecutive-loss-tolerance>5</max-consecutive-loss-tolerance>
      <max-misordering>0</max-misordering>
    </traffic-requirements>
    <member-services>ssl-1</member-services>
    <member-services>ssl-2</member-services>
  </traffic-profile>
  <traffic-profile>
    <profile-name>3</profile-name>
    <flow-spec>
      <interval>5</interval>
      <max-pkts-per-interval>10</max-pkts-per-interval>
      <max-payload-size>1500</max-payload-size>
    </flow-spec>
    <member-fwd-sublayers>afl-1</member-fwd-sublayers>
  </traffic-profile>
  <service-sub-layer>
```

```
      <service-sub-layer-list>
        <name>ssl-1</name>
        <service-rank>10</service-rank>
        <traffic-profile>2</traffic-profile>
        <service-operation-type>service-initiation
        </service-operation-type>
        <service-protection>
          <service-protection-type>none</service-protection-type>
          <sequence-number-length>long-sn</sequence-number-length>
        </service-protection>
       <incoming-type>
          <app-flow>
            <app-flow-list>app-1</app-flow-list>
          </app-flow>
        </incoming-type>
        <outgoing-type>
          <forwarding-sub-layer>
            <service-outgoing-list>
              <service-outgoing-index>0</service-outgoing-index>
              <mpls-label-stack>
                <entry>
                  <id>0</id>
                  <label>100</label>
                </entry>
              </mpls-label-stack>
              <forwarding-sub-layer>afl-1</forwarding-sub-layer>
            </service-outgoing-list>
           </forwarding-sub-layer>
        </outgoing-type>
      </service-sub-layer-list>
      <service-sub-layer-list>
        <name>ssl-2</name>
        <service-rank>10</service-rank>
        <traffic-profile>2</traffic-profile>
        <service-operation-type>service-initiation
        </service-operation-type>
        <service-protection>
          <service-protection-type>none</service-protection-type>
          <sequence-number-length>long-sn</sequence-number-length>
        </service-protection>
       <incoming-type>
          <app-flow>
            <app-flow-list>app-2</app-flow-list>
          </app-flow>
        </incoming-type>
        <outgoing-type>
          <forwarding-sub-layer>
            <service-outgoing-list>
              <service-outgoing-index>0</service-outgoing-index>
              <mpls-label-stack>
                <entry>
                  <id>0</id>
                  <label>103</label>
                </entry>
              </mpls-label-stack>
              <forwarding-sub-layer>afl-1</forwarding-sub-layer>
            </service-outgoing-list>
           </forwarding-sub-layer>
```

```
        </outgoing-type>
      </service-sub-layer-list>
      </service-sub-layer>
      <forwarding-sub-layer>
      <forwarding-sub-layer-list>
        <name>afl-1</name>
        <traffic-profile>3</traffic-profile>
        <forwarding-operation-type>impose-and-forward
        </forwarding-operation-type>
        <incoming-type>
          <service-sub-layer>
            <service-sub-layer>ssl-1</service-sub-layer>
            <service-sub-layer>ssl-2</service-sub-layer>
          </service-sub-layer>
        </incoming-type>
        <outgoing-type>
          <interface>
            <outgoing-interface>eth2</outgoing-interface>
            <mpls-label-stack>
              <entry>
                <id>0</id>
                <label>10000</label>
              </entry>
            </mpls-label-stack>
           </interface>
        </outgoing-type>
      </forwarding-sub-layer-list>
      </forwarding-sub-layer>
  </detnet>
```

Figure 5: Example B-1 DetNet XML configuration

A.3. Example B-2 JSON Service Aggregation Configuration

This illustrates the service sub-layers of DetNet. Flows 1 and 2 are aggregated into a service sub-layer of aggregated DetNet flow 1. A diagram illustrating this case is shown and then the corresponding JSON operational data follows.

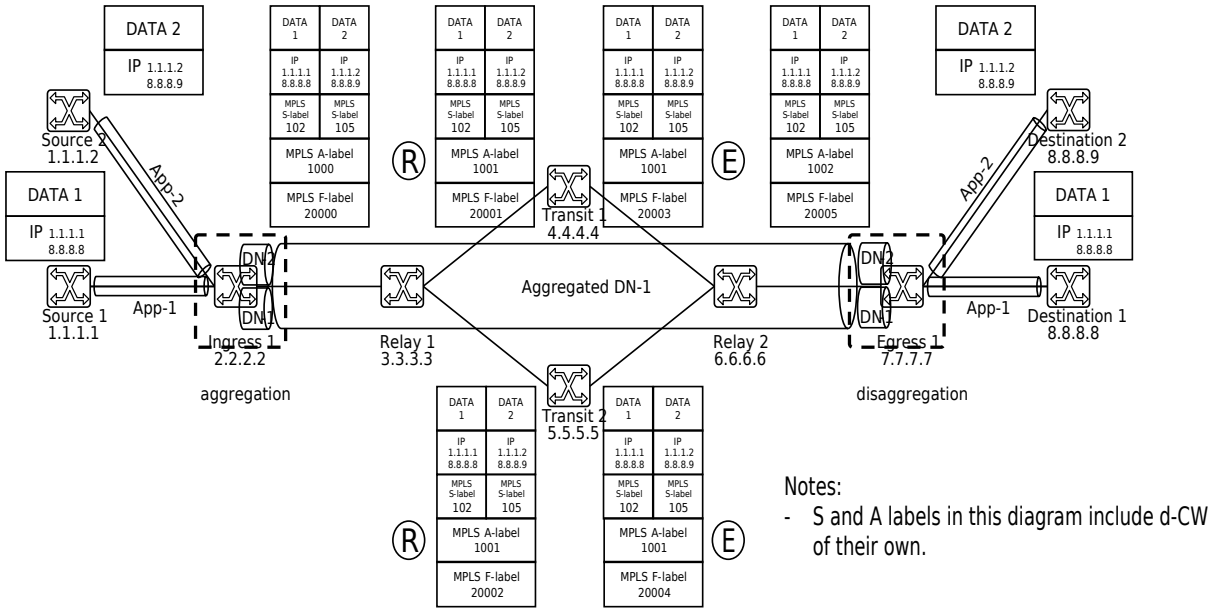


Figure 6: Case B-2 Example JSON Service Aggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "app-flows": {
      "app-flow": [
        {
          "name": "app-1",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-1",
          "traffic-profile": "1",
          "ingress": {
            "app-flow-status": "ready",
            "interface": "eth0",
            "ip-app-flow": {
```

```
              "src-ip-prefix": "1.1.1.1/32",
              "dest-ip-prefix": "8.8.8.8/32",
              "dscp": 6
            }
          }
        },
        {
          "name": "app-2",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-2",
          "traffic-profile": "1",
          "ingress": {
            "app-flow-status": "ready",
            "interface": "eth0",
            "ip-app-flow": {
              "src-ip-prefix": "1.1.1.2/32",
              "dest-ip-prefix": "8.8.8.9/32",
              "dscp": 7
            }
          }
        }
      ]
    },
    "traffic-profile": [
      {
        "profile-name": "1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 200000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
        "member-apps": [
          "app-1",
          "app-2"
        ]
      },
      {
        "profile-name": "2",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 200000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
        "member-services": [
          "ssl-1",
          "ssl-2"
        ]
      },
      {
        "profile-name": "3",
        "flow-spec": {
          "interval": 5,
```

```
          "max-pkts-per-interval": 10,
          "max-payload-size": 1500
        },
        "member-fwd-sublayers": [
          "afl-1"
        ]
      }
    ],
    "service-sub-layer": {
      "service-sub-layer-list": [
        {
          "name": "ssl-1",
          "service-rank": 10,
          "traffic-profile": "2",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-initiation",
          "incoming-type": {
            "app-flow": {
              "app-flow-list": [
                "app-1"
              ]
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "aggregation-service-sub-layer": "asl-1",
              "service-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 102
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "2",
          "service-operation-type": "service-initiation",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "incoming-type": {
            "app-flow": {
              "app-flow-list": [
                "app-2"
              ]
            }
          },
```

```
          "outgoing-type": {
            "service-sub-layer": {
              "aggregation-service-sub-layer": "asl-1",
              "service-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 105
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "asl-1",
          "service-rank": 10,
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "incoming-type": {
            "service-aggregation": {
              "service-sub-layer": [
                "ssl-1",
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 1000
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "afl-1"
                  ]
                }
              ]
            }
          }
        }
      ]
    },
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "afl-1",
```

```
          "traffic-profile": "3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 7: Example B-2 DetNet JSON Service Aggregation

A.4. Example C-1 JSON Relay Aggregation/Disaggregation Configuration

This illustrates the Relay node 1 aggregating the forwarding sub-layers of DetNet flows 1 and 2 into a forwarding sub-layer. A diagram illustrating both aggregation and disaggregation is shown and then the corresponding JSON operational data follows.

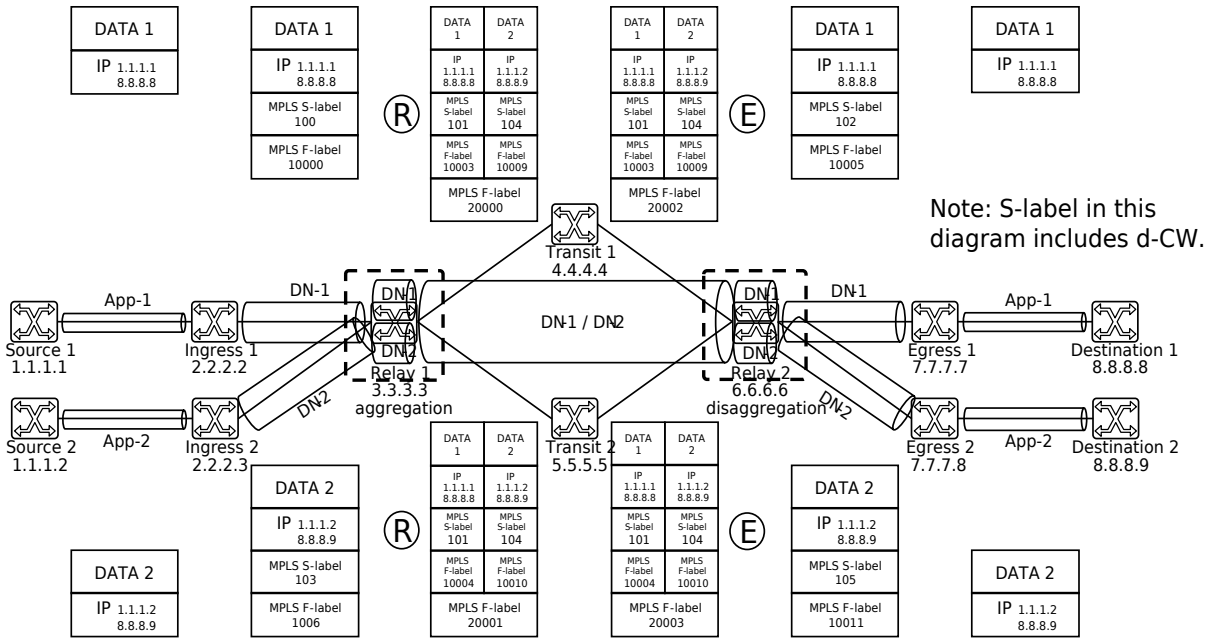


Figure 8: Case C-1 Example JSON Service Aggregation/Disaggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
        "member-services": [
          "ssl-1",
          "ssl-2"
        ]
      },
      {
        "profile-name": "pf-2",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 2,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "afl-1",
          "afl-2"
        ]
      },
      {
        "profile-name": "pf-3",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 1,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "fsl-1",
          "fsl-2",
          "fsl-3",
          "fsl-4",
          "fsl-5",
          "fsl-6"
        ]
      }
    ],
    "service-sub-layer": {
      "service-sub-layer-list": [
        {
          "name": "ssl-1",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "replication",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 100
                  }
                ]
              }
            }
          },
          "outgoing-type": {
```

```
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 101
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-2",
                    "fsl-3"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "replication",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 103
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 104
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-5",
                    "fsl-6"
```

```
                ]
              }
            ]
          }
        }
      }
    ]
  },
  "forwarding-sub-layer": {
    "forwarding-sub-layer-list": [
      {
        "name": "fsl-1",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth0",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 10000
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-1"
            ]
          }
        }
      },
      {
        "name": "fsl-2",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "impose-and-forward",
        "incoming-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-1"
            ]
          }
        },
        "outgoing-type": {
          "forwarding-sub-layer": {
            "aggregation-forwarding-sub-layer": "afl-1",
            "forwarding-label": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10003
                  }
                ]
              }
```

```
            }
          }
        }
      },
      {
        "name": "fsl-3",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "impose-and-forward",
        "incoming-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-1"
            ]
          }
        },
        "outgoing-type": {
          "forwarding-sub-layer": {
            "aggregation-forwarding-sub-layer": "afl-2",
            "forwarding-label": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10004
                  }
                ]
              }
            }
          }
        }
      },
      {
        "name": "fsl-4",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth1",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 10006
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-2"
            ]
          }
        }
      },
      {
        "name": "fsl-5",
```

```
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "aggregation-forwarding-sub-layer": "afl-1",
              "forwarding-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 10009
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "fsl-6",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "aggregation-forwarding-sub-layer": "afl-2",
              "forwarding-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 10010
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "afl-1",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "forwarding-aggregation": {
```

```
              "forwarding-sub-layer": [
                "fsl-2",
                "fsl-5"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
                  }
                ]
              }
            }
          }
        },
        {
          "name": "afl-2",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "forwarding-aggregation": {
              "forwarding-sub-layer": [
                "fsl-3",
                "fsl-6"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20001
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 9: Example C-1 DetNet JSON Relay Service Aggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "afl-1",
            "afl-2"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2",
            "fsl-3",
            "fsl-4",
            "fsl-5",
            "fsl-6"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "elimination",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
              "service-id": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 101
                    }
                  ]
                }
              }
            },
            "outgoing-type": {
```

```
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 102
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-3"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "elimination",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 104
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 105
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-6"
                  ]
                }
```

```
            ]
          }
        }
      }
    ]
  },
  "forwarding-sub-layer": {
    "forwarding-sub-layer-list": [
      {
        "name": "afl-1",
        "traffic-profile": "pf-2",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth0",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 20002
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "forwarding-disaggregation": {
            "forwarding-sub-layer": [
              "fsl-1",
              "fsl-4"
            ]
          }
        }
      },
      {
        "name": "afl-2",
        "traffic-profile": "pf-2",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth1",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 20003
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "forwarding-disaggregation": {
            "forwarding-sub-layer": [
              "fsl-2",
              "fsl-5"
            ]
          }
        }
```

```
          }
        },
        {
          "name": "fsl-1",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10003
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10004
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-3",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
```

```
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10005
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-4",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10009
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-5",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10010
                  }
                ]
```

```
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-6",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10011
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 10: Example C-1 DetNet JSON Relay Service Disaggregation

A.5. Example C-2 JSON Relay Aggregation Service Sub-Layer

This illustrates the Relay node 1 aggregating the service sub-layers of DetNet flows 1 and 2 into a forwarding sub-layer A diagram illustrating both aggregation and disaggregation is shown and then the corresponding JSON operational data follows.

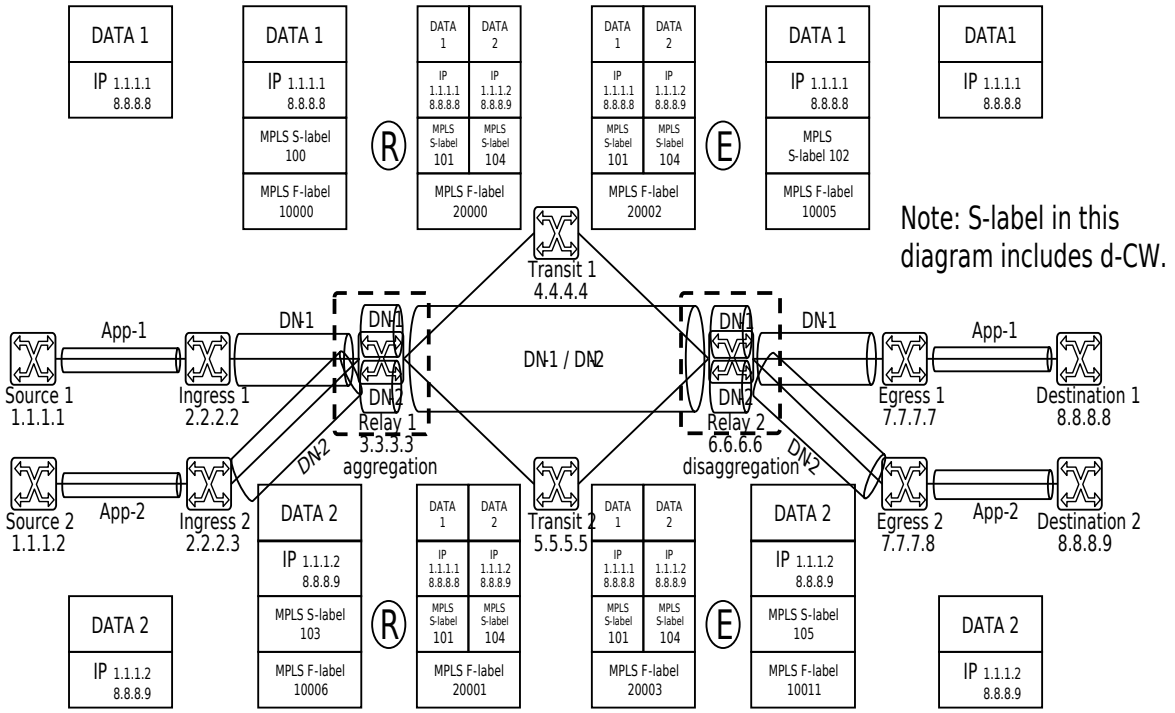


Figure 11: Case C-2 Example JSON Service Aggregation/Disaggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "afl-1",
            "afl-2"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "replication",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
              "service-id": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 100
                    }
                  ]
                }
              }
            },
            "outgoing-type": {
              "forwarding-sub-layer": {
                "service-outgoing-list": [
                  {
                    "service-outgoing-index": 0,
```

```
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 101
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "afl-1",
                  "afl-2"
                ]
              }
            ]
          }
        }
      },
      {
        "name": "ssl-2",
        "service-rank": 10,
        "traffic-profile": "pf-1",
        "service-protection": {
          "service-protection-type": "replication",
          "sequence-number-length": "long-sn"
        },
        "service-operation-type": "service-relay",
        "incoming-type": {
          "service-id": {
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 103
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "forwarding-sub-layer": {
            "service-outgoing-list": [
              {
                "service-outgoing-index": 0,
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 104
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "afl-1",
                  "afl-2"
                ]
              }
            ]
          }
```

```
          }
        }
      ]
    },
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10000
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10006
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "afl-1",
```

```
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1",
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
                  }
                ]
              }
            }
          }
        },
        {
          "name": "afl-2",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1",
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20001
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 12: Example C-2 DetNet JSON Relay Aggregation Service Sub-Layer

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "afl-1",
            "afl-2"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "elimination",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
              "service-id": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 101
                    }
                  ]
                }
              }
            },
            "outgoing-type": {
              "forwarding-sub-layer": {
                "service-outgoing-list": [
                  {
                    "service-outgoing-index": 0,
```

```
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 102
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "fsl-1"
                ]
              }
            ]
          }
        }
      },
      {
        "name": "ssl-2",
        "service-rank": 10,
        "traffic-profile": "pf-1",
        "service-protection": {
          "service-protection-type": "elimination",
          "sequence-number-length": "long-sn"
        },
        "service-operation-type": "service-relay",
        "incoming-type": {
          "service-id": {
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 104
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "forwarding-sub-layer": {
            "service-outgoing-list": [
              {
                "service-outgoing-index": 0,
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 105
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "fsl-2"
                ]
              }
            ]
          }
        }
      }
```

```
      ]
    },
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "afl-1",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20002
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1",
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "afl-2",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20003
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1",
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-1",
```

```
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10005
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10011
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 13: Example C-2 DetNet JSON Relay Disaggregation Service Sub-Layer

A.6. Example C-3 JSON Relay Service Sub-Layer Aggregation/ Disaggregation

This illustrates the Relay node 1 aggregating the service sub-layers of DetNet flows 1 and 2 into a service sub-layer of Aggregated DetNet flow 1. It also illustrates the Relay node 1 disaggregating the aggregated DetNet flow 1 into the DetNet flows 1 and 2 service sub-layers. A diagram illustrating both aggregation and disaggregation is shown and then the corresponding JSON operational data follows.

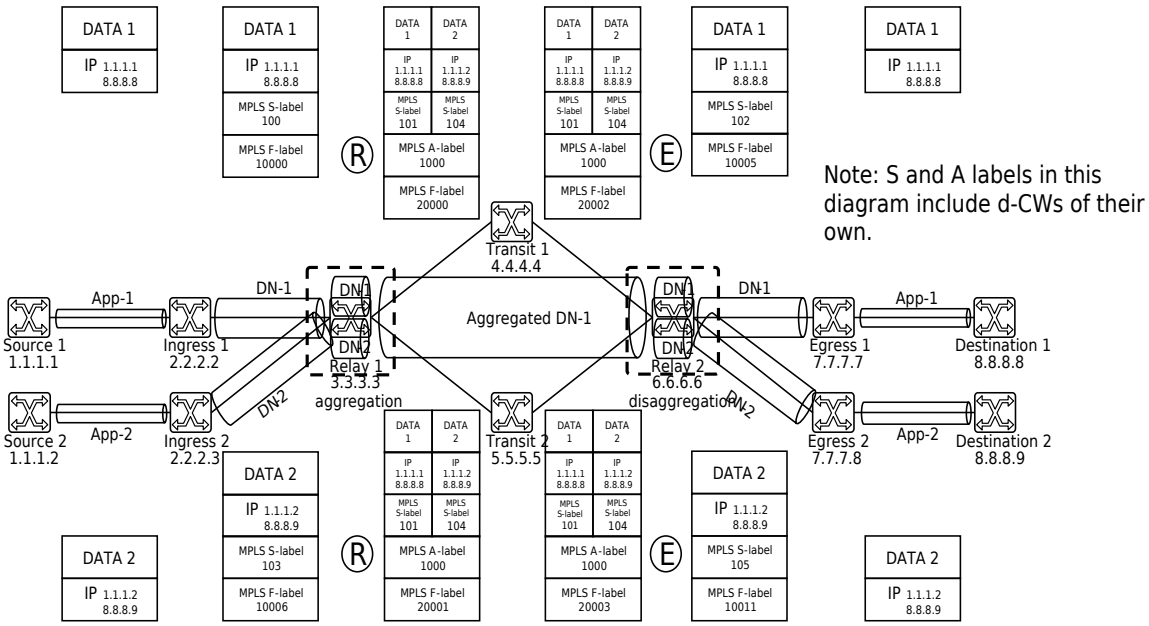


Figure 14: Case C-3 Example JSON Service Aggregation/Disaggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "traffic-requirements": {
            "min-bandwidth": "200000000",
            "max-latency": 100000000,
            "max-latency-variation": 100000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "member-services": [
            "asl-1"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2"
          ]
        },
        {
          "profile-name": "pf-4",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-3",
            "fsl-4"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "none",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
              "service-id": {
                "mpls-label-stack": {
```

```
                "entry": [
                  {
                    "id": 0,
                    "label": 100
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "aggregation-service-sub-layer": "asl-1",
              "service-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 101
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 103
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "aggregation-service-sub-layer": "asl-1",
              "service-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 104
                    }
                  ]
                }
```

```
            }
          }
        }
      },
      {
        "name": "asl-1",
        "service-rank": 10,
        "traffic-profile": "pf-2",
        "service-protection": {
          "service-protection-type": "replication",
          "sequence-number-length": "long-sn"
        },
        "service-operation-type": "service-initiation",
        "incoming-type": {
          "service-aggregation": {
            "service-sub-layer": [
              "ssl-1",
              "ssl-2"
            ]
          }
        },
        "outgoing-type": {
          "forwarding-sub-layer": {
            "service-outgoing-list": [
              {
                "service-outgoing-index": 0,
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 1000
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "fsl-3",
                  "fsl-4"
                ]
              }
            ]
          }
        }
      }
    ]
  },
  "forwarding-sub-layer": {
    "forwarding-sub-layer-list": [
      {
        "name": "fsl-1",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth0",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
```

```
                    "label": 10000
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10006
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-3",
          "traffic-profile": "pf-4",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
```

```
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-4",
          "traffic-profile": "pf-4",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20001
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 15: Example C-3 DetNet JSON Relay Service Sub-Layer Aggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "traffic-requirements": {
            "min-bandwidth": "200000000",
            "max-latency": 100000000,
            "max-latency-variation": 100000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "member-services": [
            "asl-1"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-3",
            "fsl-4"
          ]
        },
        {
          "profile-name": "pf-4",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "none",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
              "service-id": {
                "mpls-label-stack": {
```

```
                "entry": [
                  {
                    "id": 0,
                    "label": 101
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 102
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-3"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 104
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
```

```
                    {
                      "id": 0,
                      "label": 105
                    }
                  ]
                },
                "forwarding-sub-layer": [
                  "fsl-4"
                ]
              }
            ]
          }
        }
      },
      {
        "name": "asl-1",
        "service-rank": 10,
        "traffic-profile": "pf-2",
        "service-protection": {
          "service-protection-type": "elimination",
          "sequence-number-length": "long-sn"
        },
        "service-operation-type": "service-termination",
        "incoming-type": {
          "service-id": {
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 1000
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "service-disaggregation": {
            "service-sub-layer": [
              "ssl-1",
              "ssl-2"
            ]
          }
        }
      }
    ]
  },
  "forwarding-sub-layer": {
    "forwarding-sub-layer-list": [
      {
        "name": "fsl-1",
        "traffic-profile": "pf-4",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth0",
            "mpls-label-stack": {
              "entry": [
                {
```

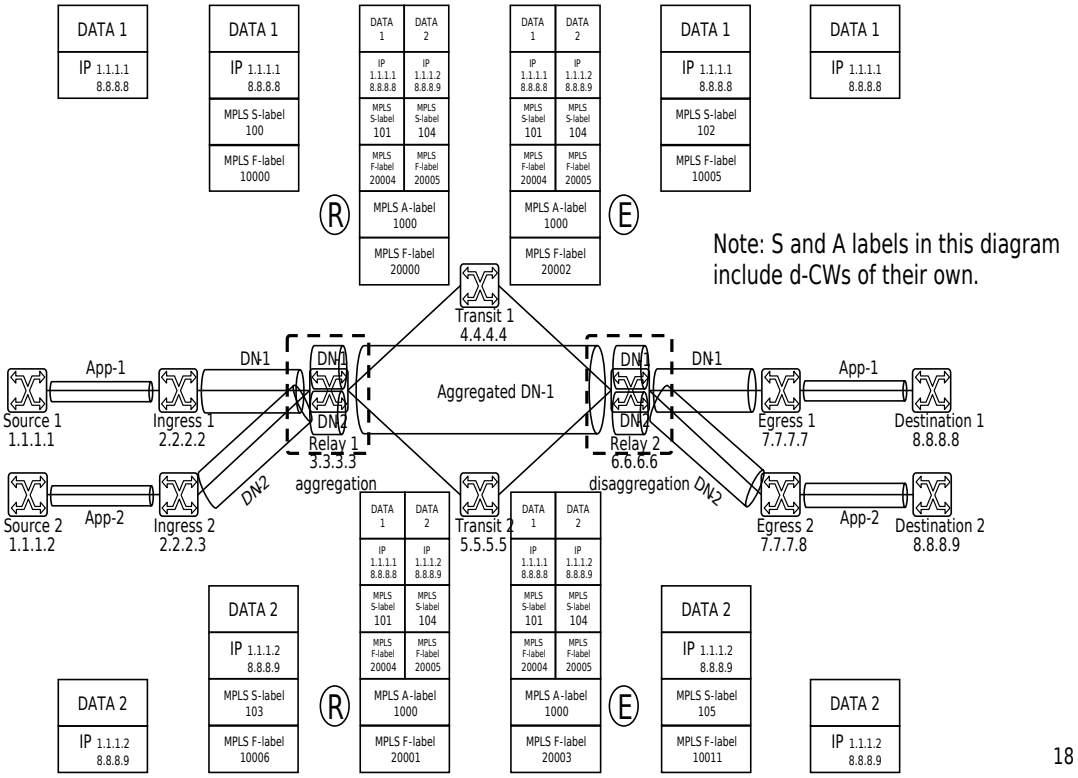
```
                  "id": 0,
                  "label": 20002
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "asl-1"
            ]
          }
        }
      },
      {
        "name": "fsl-2",
        "traffic-profile": "pf-4",
        "forwarding-operation-type": "pop-and-lookup",
        "incoming-type": {
          "forwarding-id": {
            "interface": "eth1",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 20003
                }
              ]
            }
          }
        },
        "outgoing-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "asl-1"
            ]
          }
        }
      },
      {
        "name": "fsl-3",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "impose-and-forward",
        "incoming-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-1"
            ]
          }
        },
        "outgoing-type": {
          "interface": {
            "outgoing-interface": "eth2",
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
```

```
                    "label": 10005
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-4",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10011
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 16: Example C-3 DetNet JSON Relay Service Sub-Layer Disaggregation

A.7. Example C-4 JSON Relay Service Sub-Layer Aggregation/ Disaggregation

This illustrates the Relay node 1 aggregating the forwarding sub-layers of DetNet flow 1 and 2 into a service sub-layer of Aggregated DetNet flow 1. This also illustrates the Relay node 1 disaggregating the service sub-layer of Aggregated DetNet flow 1 to forwarding sub-layers of DetNet flow 1 and 2. A diagram illustrating both aggregation and disaggregation is shown and then the corresponding JSON operational data follows.



18

Figure 17: Case C-4 Example JSON Service Aggregation/Disaggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "traffic-requirements": {
            "min-bandwidth": "200000000",
            "max-latency": 100000000,
            "max-latency-variation": 100000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "member-services": [
            "asl-1"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2",
            "fsl-3",
            "fsl-4"
          ]
        },
        {
          "profile-name": "pf-4",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-5",
            "fsl-6"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "none",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
```

```
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 100
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 101
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-3"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 103
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
```

```
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 104
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-4"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "asl-1",
          "service-rank": 10,
          "traffic-profile": "pf-2",
          "service-protection": {
            "service-protection-type": "replication",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-initiation",
          "incoming-type": {
            "forwarding-aggregation": {
              "forwarding-sub-layer": [
                "fsl-3",
                "fsl-4"
              ]
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 1000
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-5",
                    "fsl-6"
                  ]
                }
              ]
            }
          }
        }
      ]
    },
    "forwarding-sub-layer": {
```

```
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10000
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10006
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-3",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
```

```
              "ssl-1"
            ]
          }
        },
        "outgoing-type": {
          "service-aggregation": {
            "aggregation-service-sub-layer": "asl-1",
            "optional-forwarding-label": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20004
                  }
                ]
              }
            }
          }
        }
      },
      {
        "name": "fsl-4",
        "traffic-profile": "pf-3",
        "forwarding-operation-type": "impose-and-forward",
        "incoming-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "ssl-2"
            ]
          }
        },
        "outgoing-type": {
          "service-aggregation": {
            "aggregation-service-sub-layer": "asl-1",
            "optional-forwarding-label": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20005
                  }
                ]
              }
            }
          }
        }
      },
      {
        "name": "fsl-5",
        "traffic-profile": "pf-4",
        "forwarding-operation-type": "impose-and-forward",
        "incoming-type": {
          "service-sub-layer": {
            "service-sub-layer": [
              "asl-1"
            ]
          }
        },
```

```
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-6",
          "traffic-profile": "pf-4",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20001
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 18: Example C-4 DetNet JSON Relay Service Sub-Layer Aggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "traffic-requirements": {
          "min-bandwidth": "100000000",
          "max-latency": 100000000,
          "max-latency-variation": 100000000,
          "max-loss": 2,
          "max-consecutive-loss-tolerance": 5,
          "max-misordering": 0
        },
```

```
          "member-services": [
            "ssl-1",
            "ssl-2"
          ]
        },
        {
          "profile-name": "pf-2",
          "traffic-requirements": {
            "min-bandwidth": "200000000",
            "max-latency": 100000000,
            "max-latency-variation": 100000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "member-services": [
            "asl-1"
          ]
        },
        {
          "profile-name": "pf-3",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 1,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-3",
            "fsl-4",
            "fsl-5",
            "fsl-6"
          ]
        },
        {
          "profile-name": "pf-4",
          "flow-spec": {
            "interval": 125,
            "max-pkts-per-interval": 2,
            "max-payload-size": 1518
          },
          "member-fwd-sublayers": [
            "fsl-1",
            "fsl-2"
          ]
        }
      ],
      "service-sub-layer": {
        "service-sub-layer-list": [
          {
            "name": "ssl-1",
            "service-rank": 10,
            "traffic-profile": "pf-1",
            "service-protection": {
              "service-protection-type": "none",
              "sequence-number-length": "long-sn"
            },
            "service-operation-type": "service-relay",
            "incoming-type": {
```

```
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 101
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 102
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-5"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-profile": "pf-1",
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-relay",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 104
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "service-outgoing-list": [
                {
                  "service-outgoing-index": 0,
```

```
                  "mpls-label-stack": {
                    "entry": [
                      {
                        "id": 0,
                        "label": 105
                      }
                    ]
                  },
                  "forwarding-sub-layer": [
                    "fsl-6"
                  ]
                }
              ]
            }
          }
        },
        {
          "name": "asl-1",
          "service-rank": 10,
          "traffic-profile": "pf-2",
          "service-protection": {
            "service-protection-type": "elimination",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-termination",
          "incoming-type": {
            "service-id": {
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 1000
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-disaggregation": {
              "forwarding-sub-layer": [
                "fsl-3",
                "fsl-4"
              ]
            }
          }
        }
      ]
    },
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-4",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
```

```
                "entry": [
                  {
                    "id": 0,
                    "label": 20002
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-4",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20003
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "asl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-3",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20004
                  }
                ]
              }
            }
          }
```

```
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          }
        },
        {
          "name": "fsl-4",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20005
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          }
        },
        {
          "name": "fsl-5",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-1"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10005
                  }
                ]
              }
            }
          }
```

```
        },
        {
          "name": "fsl-6",
          "traffic-profile": "pf-3",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "service-sub-layer": {
              "service-sub-layer": [
                "ssl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10011
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 19: Example C-4 DetNet JSON Relay Service Sub-Layer Disaggregation

A.8. Example D-1 JSON Transit Forwarding Sub-Layer Aggregation/Disaggregation

This illustrates the Transit node 1 aggregating the forwarding sub-layers of DetNet flow 1 and 2 into a forwarding sub-layer. This also illustrates a Transit node 1 disaggregating a forwarding sub-layer into DetNet flow 1 and 2 forwarding sub-layers.

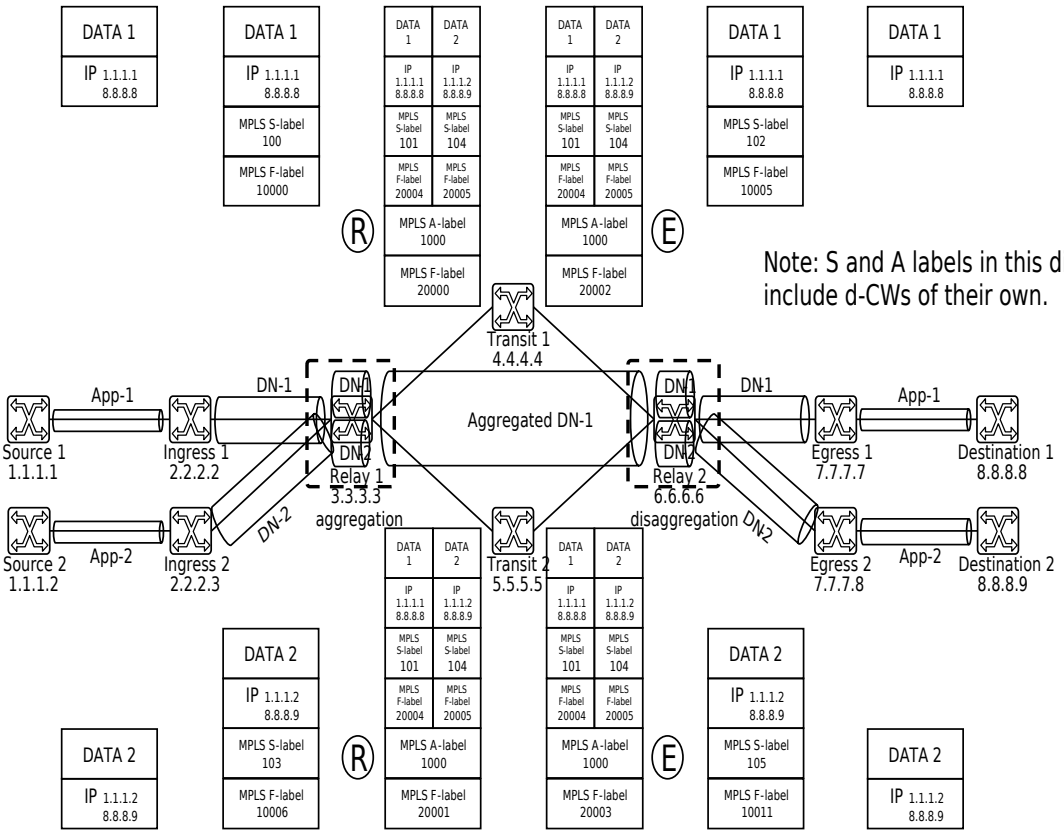


Figure 20: Case D-1 Example Service Aggregation/Disaggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 1,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "fsl-1",
          "fsl-2"
```

```
        ]
      },
      {
        "profile-name": "pf-2",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 2,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "afl-1"
        ]
      }
    ],
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-1",
          "forwarding-operation-type": "pop-impose-and-forward",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth0",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10000
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "aggregation-forwarding-sub-layer": "afl-1",
              "forwarding-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 10002
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-1",
          "forwarding-operation-type": "pop-impose-and-forward",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
```

```
                  {
                    "id": 0,
                    "label": 10004
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-sub-layer": {
              "aggregation-forwarding-sub-layer": "afl-1",
              "forwarding-label": {
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 0,
                      "label": 10006
                    }
                  ]
                }
              }
            }
          }
        },
        {
          "name": "afl-1",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "impose-and-forward",
          "incoming-type": {
            "forwarding-aggregation": {
              "forwarding-sub-layer": [
                "fsl-1",
                "fsl-2"
              ]
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20000
                  }
                ]
              }
            }
          }
        }
      ]
    }
  }
}
```

Figure 21: Example D-1 DetNet JSON Relay Service Sub-Layer Aggregation

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-12-18T23:59:00Z"
        }
      }
    ]
  },
  "ietf-detnet:detnet": {
    "traffic-profile": [
      {
        "profile-name": "pf-1",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 1,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "fsl-1",
          "fsl-2"
```

```
        ]
      },
      {
        "profile-name": "pf-2",
        "flow-spec": {
          "interval": 125,
          "max-pkts-per-interval": 2,
          "max-payload-size": 1518
        },
        "member-fwd-sublayers": [
          "afl-1"
        ]
      }
    ],
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-profile": "pf-1",
          "forwarding-operation-type": "swap-and-forward",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10002
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth3",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10003
                  }
                ]
              }
            }
          }
        },
        {
          "name": "fsl-2",
          "traffic-profile": "pf-1",
          "forwarding-operation-type": "swap-and-forward",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
```

```
                    "label": 10006
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "interface": {
              "outgoing-interface": "eth2",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 10007
                  }
                ]
              }
            }
          }
        },
        {
          "name": "afl-1",
          "traffic-profile": "pf-2",
          "forwarding-operation-type": "pop-and-lookup",
          "incoming-type": {
            "forwarding-id": {
              "interface": "eth1",
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 20001
                  }
                ]
              }
            }
          },
          "outgoing-type": {
            "forwarding-disaggregation": {
              "forwarding-sub-layer": [
                "fsl-1",
                "fsl-2"
              ]
            }
          }
        }
      ]
    }
  }
}
```

Figure 22: Example D-1 DetNet JSON Relay Service Sub-Layer Disaggregation

Authors' Addresses

Xuesong Geng
Huawei Technologies
Email: gengxuesong@huawei.com

Mach(Guoyi) Chen
Huawei Technologies
Email: mach.chen@huawei.com

Yeoncheol Ryoo
ETRI
Email: dbduscjf@etri.re.kr

Don Fedyk
LabN Consulting, L.L.C.
Email: dfedyk@labn.net

Reshad Rahman
Individual
Email: reshad@yahoo.com

Zhenqiang Li
China Mobile
Email: lizhenqiang@chinamobile.com