DetNet Configuration YANG Model

draft-ietf-detnet-yang-07

Xuesong Geng (gengxuesong@huawei.com)

Mach Chen (mach.chen@huawei.com)

Yeoncheol Ryoo (dbduscjf@etri.re.kr)

Don Fedyk(dfedyk@labn.net)

Zhenqiang Li (lizhengqiang@chinamobile.com)

Reshad Rahman(rrahman@cisco.com)

Status

- Merged two Models from Last draft
  - Some terminology alignment.
  - Agreed to work the remaining changes into the single model.
  - Now Supports the flow model attributes
- Need to Publish Yanglint config for sample configurations
  - This is working for the model but it is not in the draft.
- Plan to review Model and add Service Sub-layer aggregation
  - Forwarding Sub layer aggregation falls out from the model.
- Need to check terminology for consistency and clarity.
- Need to confirm data plane match data plane drafts.

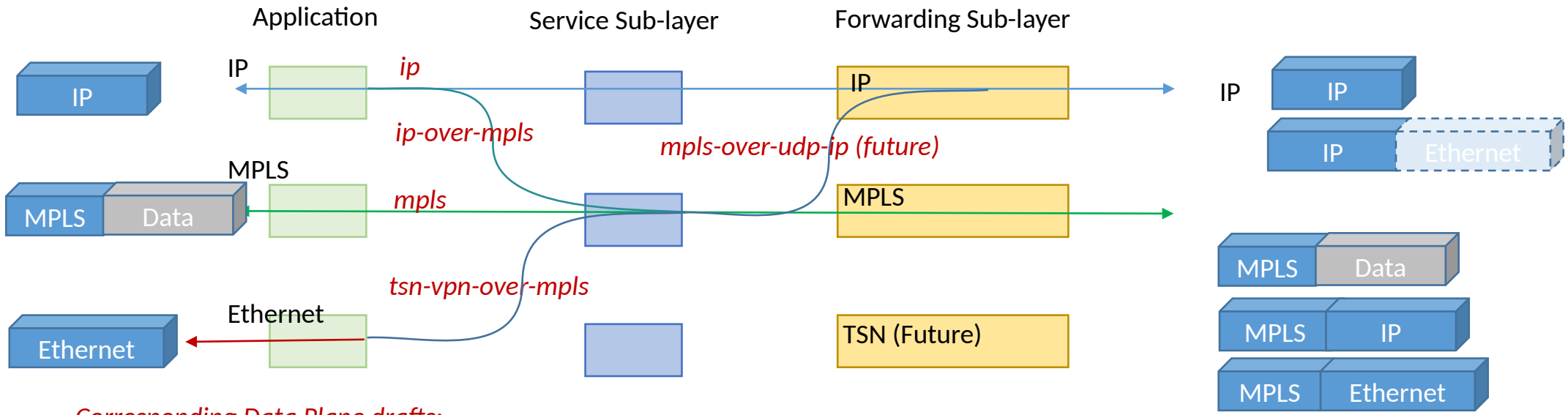
History

- Version 00: accepted as a WG document after IETF 102
- Version 01: *ietf-detnet-topology-yang* is defined independently
- Version 02: updated following the feedback from IETF103
  - Add ‘Sequence Number Generation’
    - OAM considerations
  - Add ‘DetNet Service Decapsulation’
  - Add ‘DetNet Transport Tunnel Decapsulation’
- Version 03: DetNet Configuration Structure Update in IETF104 and IETF105
- Version 04 :
  - Modify the scope of DetNet YANG Model
- Version 05/06:
  - Two YANG Models Discussion -> Comparison
- Version 07
  - Merging Models. Some terminology alignment.

WG Call Meeting Every Week

Scenarios Covered by DetNet YANG Model

(w/o Aggregation)



Corresponding Data Plane drafts:
draft-ietf-detnet-ip-06
draft-ietf-detnet-ip-over-mpls-06
draft-ietf-detnet-mpls-07
draft-ietf-detnet-mpls-over-udp-ip-06 (Partial)
draft-ietf-detnet-tsn-vpn-over-mpls-03 (Partial)
draft-ietf-detnet-mpls-over-tsn-03(Not yet)
draft-ietf-detnet-ip-over-tsn-03(Not yet)

Not shown Ethernet or other Tunnels as be underlay

Flow Model Attributes Supported by YANG

App-flow, DetNet flow and DetNet service

draft-ietf-detnet-flow-information-model

App-flow

Characteristics

- FlowID: unique (manag.) ID
- FlowType: Eth, MPLS, IP
- *DataFlowSpecification:*
  src/dst-addr, label, VLAN, etc.
- *TrafficSpecification:*
  interval, pckt-size, max-packet
- FlowEndPoints: Src, Dst(s)
- FlowRank
- FlowStatus

Requirements

- *FlowRequirements:*
  MinBW, PD, PDV, Loss, etc.
- FlowBiDir

Service Requirements similar to e.g., 802.1Qcc Attributes like UserToNetworkRequirements

DetNet flow

Characteristics

- DnFlowID: unique (manag.) ID
- DnPayloadType: Eth, MPLS, IP
- DnFlowFormat: MPLS, IP
- *DnFlowSpecification:*
  Label, 6-tuple
- *DnTrafficSpecification*
  interval, pckt-size, max-packet
- DnFlowEndPoints: Ingress, Egress(s)
- DnFlowRank
- DnFlowStatus

Requirements

- *DnFlowRequirements:*
  MinBW, MaxLatency, MaxLatencyVariation, MaxLoss, MaxConsecutiveLossTolerance, MaxMisordering
- DnFlowBiDir

DN Service

- DnServiceID: unique (manag.) ID
- DnServiceDeliveryType: Eth, MPLS, IP
- DnServiceConnectivity: p2p, p2mp
- DnServiceRank
- *DnServiceDeliveryProfile:*
  MaxBW, MaxLatency, MaxLatencyVariation, MaxLoss, MaxConsecutiveLossTolerance, MaxMisordering
- DnServiceBiDir
- DnServiceStatus

A DetNet flow contains one or more App-flows (N:1 mapping).

A DetNet service supports one or more DetNet-flows (M:1 mapping).

Relating the DetNet flows to the YANG Model

From : https://tools.ietf.org/html/draft-ietf-detnet-mpls-05

> "From the data plane App-flow identification at a DetNet service sub-layer is realized by an S-Label."

From : https://tools.ietf.org/html/draft-ietf-detnet-flow-information-model-10

> "A DetNet flow includes one or more App-flow(s) as payload."

- When aggregation is added to the YANG model, with A-Labels in the MPLS case, we will have DetNet Flows.

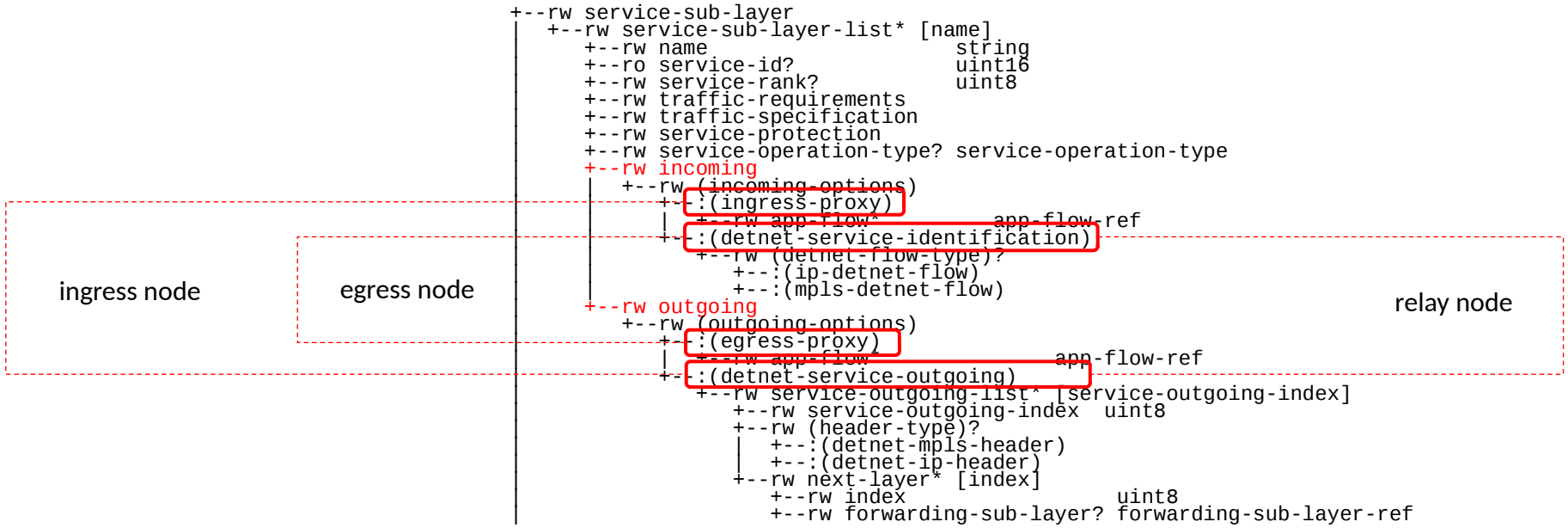
Updated YANG Model tree : App-flow

```
module: ietf-detnet-config
  +--rw detnet
     +--rw app-flows
     |  +--rw app-flow* [name]
     |     +--rw name                        string
     |     +--ro app-id?                     uint16
     |     +--rw app-flow-bidir-congruent? boolean
     |     +--rw outgoing-service?           service-sub-layer-ref
     |     +--rw incoming-service?           service-sub-layer-ref
     |     +--rw traffic-requirements
     |     +--rw traffic-specification
     |     +--rw ingress
     |     |  +--rw name?                      string
     |     |  +--ro app-flow-status?           identityref
     |     |  +--rw interface?                 if:interface-ref
     |     |  +--rw (data-flow-type)?
     |     |     +--:(tsn-app-flow)
     |     |     +--:(ip-app-flow)
     |     |     +--:(mpls-app-flow)
     |     +--rw egress
     |        +--rw name?     string
     |        +--rw (application-type)?
     |           +--:(ethernet)
     |           +--:(ip-mpls)
     |              +--rw ip-mpls
     |                 +--rw (next-hop-options)
     |                    +--:(simple-next-hop)
     |                    |  +--rw outgoing-interface? if:interface-ref
     |                    |  +--rw (flow-type)?
     |                    |     +--:(ip)
     |                    |     +--:(mpls)
     |                    +--:(next-hop-list)
     |                       +--rw next-hop-list
     |                          +--rw next-hop* [hop-index]
     |                             +--rw hop-index           uint8
     |                             +--rw outgoing-interface? if:interface-ref
     |                             +--rw (flow-type)?
     |                                +--:(ip)
     |                                +--:(mpls)
```

Application ingress and related outgoing Detnet service

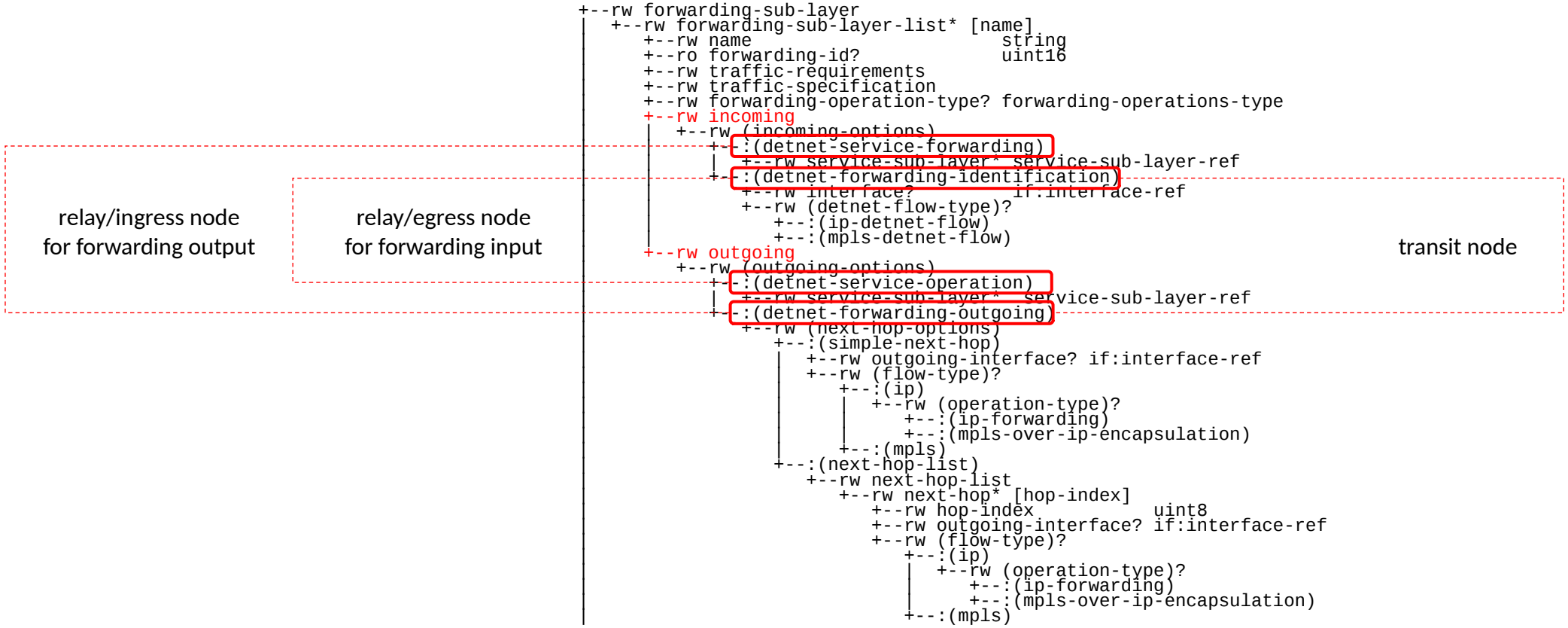
Application egress and related incoming Detnet service

Updated YANG Model tree : Service Sub-layer

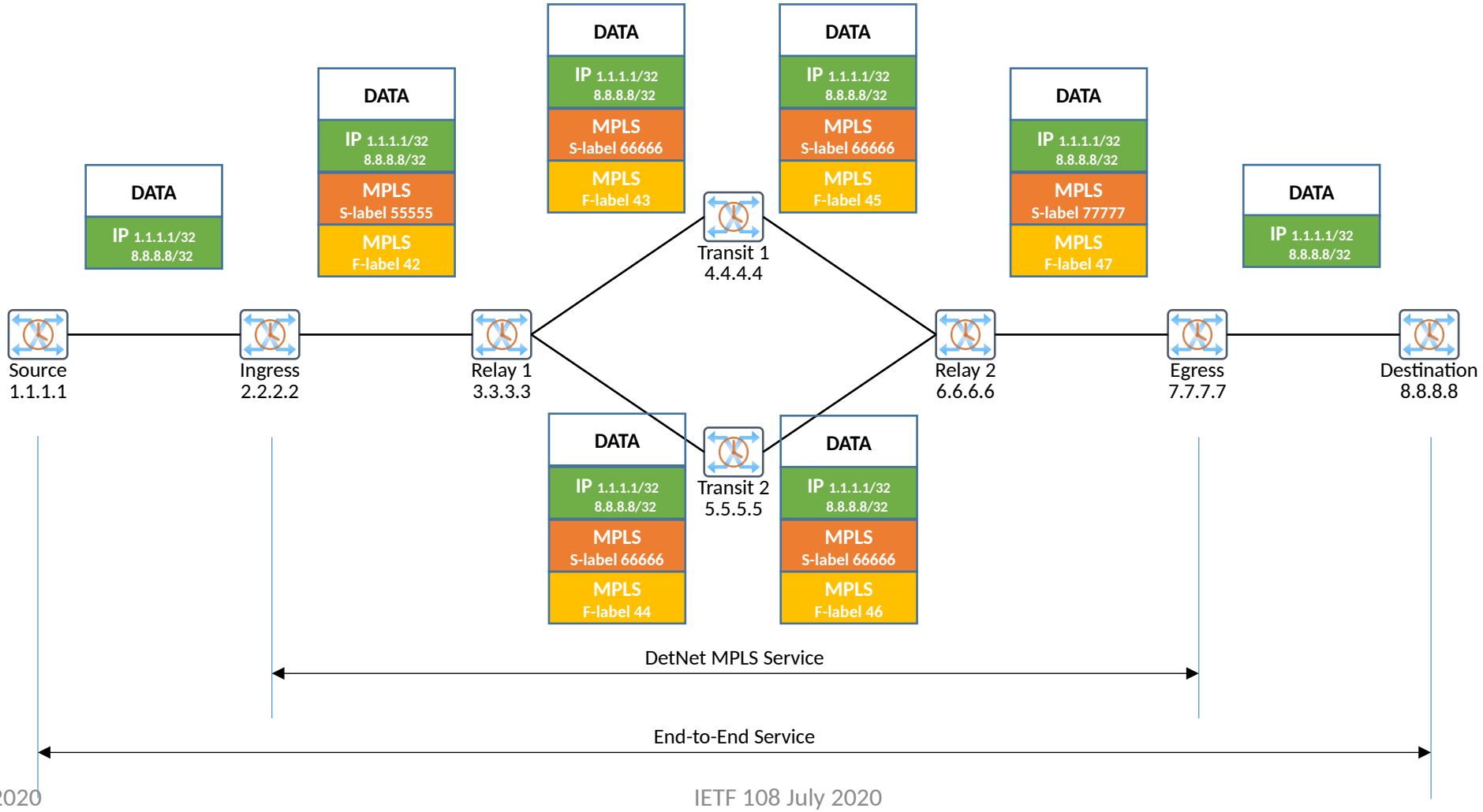



```
+--rw service-sub-layer
|  +--rw service-sub-layer-list* [name]
|     +--rw name                          string
|     +--ro service-id?                   uint16
|     +--rw service-rank?                 uint8
|     +--rw traffic-requirements
|     +--rw traffic-specification
|     +--rw service-protection
|     +--rw service-operation-type? service-operation-type
|     +--rw incoming
|     |  +--rw (incoming-options)
|     |     +--:(ingress-proxy)
|     |     |  +--rw app-flow*          app-flow-ref
|     |     +--:(detnet-service-identification)
|     |        +--rw (detnet-flow-type)?
|     |           +--:(ip-detnet-flow)
|     |           +--:(mpls-detnet-flow)
|     +--rw outgoing
|        +--rw (outgoing-options)
|           +--:(egress-proxy)
|           |  +--rw app-flow*          app-flow-ref
|           +--:(detnet-service-outgoing)
|              +--rw service-outgoing-list* [service-outgoing-index]
|                 +--rw service-outgoing-index  uint8
|                 +--rw (header-type)?
|                 |  +--:(detnet-mpls-header)
|                 |  +--:(detnet-ip-header)
|                 +--rw next-layer* [index]
|                    +--rw index                uint8
|                    +--rw forwarding-sub-layer? forwarding-sub-layer-ref
```

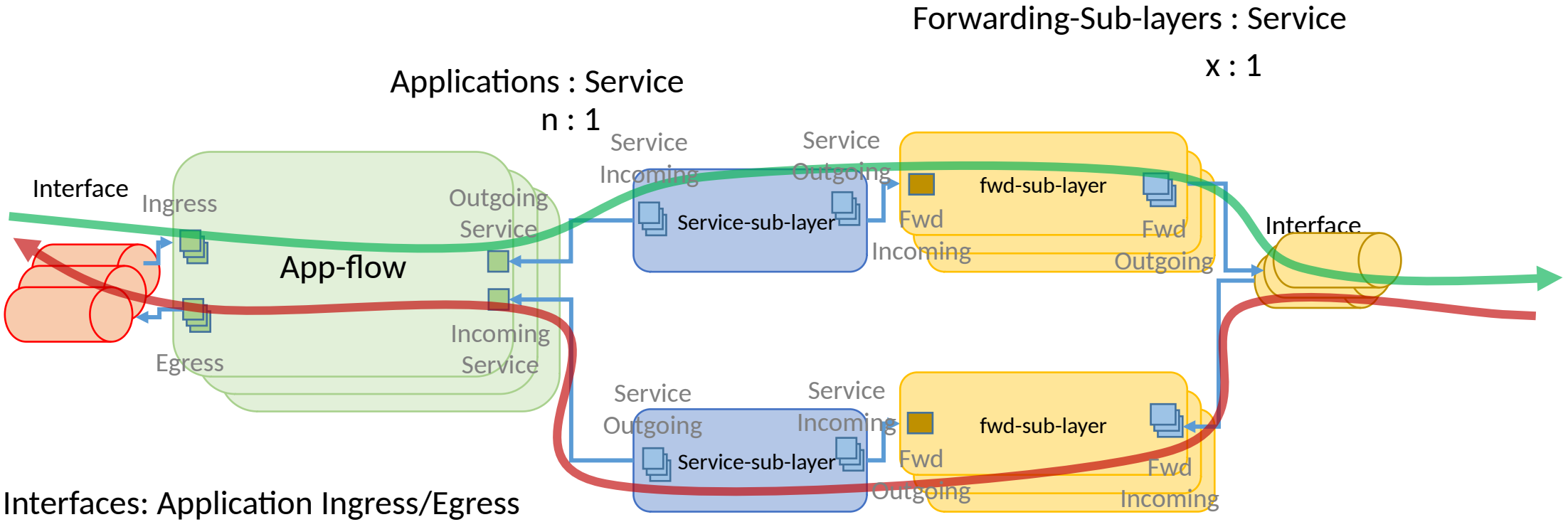
Updated YANG Model tree : Forwarding Sub-layer



Example : IP App-flow, DetNet MPLS (bidirectional)



Edge Node


Forwarding-Sub-layers : Service
x : 1
Applications : Service
n : 1
Interface
Ingress
Egress
App-flow
Outgoing
Service
Incoming
Service
Service
Incoming
Service
Outgoing
Service-sub-layer
Service
Outgoing
Service
Incoming
Service-sub-layer
fwd-sub-layer
Fwd
Incoming
Fwd
Outgoing
fwd-sub-layer
Fwd
Outgoing
Fwd
Incoming
Interface
Interfaces: Application Ingress/Egress
k : 1
DetNet Flows : Service
m : 1
Interfaces: Forwarding-Sub-layers
y : 1

Yanglint JSON sample configuration : edge node

```
> data -t config -f json ingress_node-bidir-app.xml
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd"
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd"
      }
    ]
  },
  "ietf-detnet-config:detnet": {
    "app-flows": {
      "app-flow": [
        {
          "name": "app-1",
          "app-flow-bidir-congruent": true,
          "outgoing-service": "ssl-1",
          "incoming-service": "ssl-2",
          "ingress": {
            "name": "port1",
            "interface": "eth0",
            "src-ip-prefix": "1.1.1.1/32",
            "dest-ip-prefix": "8.8.8.8/32",
            "traffic-class": 40
          },
          "egress": {
            "name": "port1",
            "ip-mpls": {
              "outgoing-interface": "eth0"
            }
          }
        }
      ]
    },
```

```
    "service-sub-layer": {
      "service-sub-layer-list": [
        {
          "name": "ssl-1",
          "service-rank": 10,
          "traffic-requirements": {
            "min-bandwidth": "100000000",
            "max-latency": 100000000,
            "max-latency-variation": 2000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-initiation",
          "incoming": {
            "app-flow": [
              "app-1"
            ]
          },
          "outgoing": {
            "service-outgoing-list": [
              {
                "service-outgoing-index": 1,
                "mpls-label-stack": {
                  "entry": [
                    {
                      "id": 1,
                      "label": 55555,
                      "traffic-class": 5
                    }
                  ]
                },
                "next-layer": [
                  {
                    "index": 0,
                    "forwarding-sub-layer": "fsl-1"
                  }
                ]
              }
            ]
          }
        },
```

```
        {
          "name": "ssl-2",
          "service-rank": 10,
          "traffic-requirements": {
            "min-bandwidth": "100000000",
            "max-latency": 100000000,
            "max-latency-variation": 2000000,
            "max-loss": 2,
            "max-consecutive-loss-tolerance": 5,
            "max-misordering": 0
          },
          "service-protection": {
            "service-protection-type": "none",
            "sequence-number-length": "long-sn"
          },
          "service-operation-type": "service-termination",
          "incoming": {
            "label": 55555
          },
          "outgoing": {
            "app-flow": [
              "app-1"
            ]
          }
        }
      ]
    },
```

```
    "forwarding-sub-layer": {
      "forwarding-sub-layer-list": [
        {
          "name": "fsl-1",
          "traffic-specification": {
            "interval": 5,
            "max-packets-per-interval": 10,
            "max-payload-size": 1500
          },
          "forwarding-operation-type": "impose-and-forward",
          "incoming": {
            "service-sub-layer": [
              "ssl-1"
            ]
          },
          "outgoing": {
            "outgoing-interface": "eth1"
          }
        },
        {
          "name": "fsl-2",
          "traffic-specification": {
            "interval": 5,
            "max-packets-per-interval": 10,
            "max-payload-size": 1500
          },
          "forwarding-operation-type": "pop-and-lookup",
          "incoming": {
            "interface": "eth1",
            "label": 42
          },
          "outgoing": {
            "service-sub-layer": [
              "ssl-2"
            ]
          }
        }
      ]
    }
  }
}
```

Plan

- Comments and reviewing the current Model.
  - Terminology consistency
  - Adding Service Sub-layer Aggregation.
  - Double check Flow Model consistency
  - Continuing with Yanglint validation of the sample configuration.
- General clean up
  - Clean up YANG
  - Address Nits for draft
  - Include Yanglint xml sample configuration

Thanks
Questions/Comments?