DetNet Service Model
draft-varga-detnet-service-model-00

Abstract

   This document describes the service model for scenarios requiring
   deterministic / time sensitive networking.

Status of This Memo

Copyright Notice

Table of Contents

1. **Introduction**

   Deterministic Networking service provides a capability to carry
   specified data flow, whether unicast or multicast, for an application
   with constrained requirements towards network performance, e.g. low
   packet loss rate and/or latency. During the discussion of detnet
   use-cases, detnet architecture and various related networking
   scenarios several confusions have been arrised due to different
   service model interpretations. This document defines service
   reference points, service components and proposes naming for service
   scenarios to achieve common understanding of the detnet service
   model.

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

The lowercase forms with an initial capital "Must", "Must Not",
"Shall", "Shall Not", "Should", "Should Not", "May", and "Optional"
in this document are to be interpreted in the sense defined in
[RFC2119], but are used where the normative behavior is defined in
documents published by SDOs other than the IETF.

3. Terminology and Definitions

Additional terms to [draft-data-plane] and [draft-arch] used/
described in this draft .

App-flow: Data flow between the applications requiring deterministic
   transport.

DetLink: Direct link between two entities (node/end-system) used for
   deterministic transport.

DetNet AC: Attachment Circuit of a DetNet transport service for a
   DetNet-flow.

DetNet-flow: Data flow requiring deterministic transport between two
   DetNet-UNIs.

DetNet-UNI: UNI of an Edge/Relay node to provide deterministic
   service for a connected node/end-system.

DetNetwork: Transport network between DetNet-UNIs.

Native AC: Attachment Circuit of a DetNet transport service for an
   App-flow.

4. End-systems connected to DetNet

Deterministic transport is required by time/loss sensitive
application(s) running on an End-system during communication with its
peer(s). Such a data exchange has various requirements on delay and/
or loss parameters. The native data flow between the source/sink
End-Systems is called as application-flow (app-flow) as shown in
Figure 1. The traffic characteristics of an app-flow can be CBR or
VBR and can have L1 or L2 or L3 format (e.g., TDM, Ethernet, IP).

[Note: Interworking function for L1 application-flows is out-of-scope
in this document and therefore not depicted on figures.]

```
     +-------------+                   +-------------+
     | Application |<---native data flow--->| Application |
     |-------------|   (application-flow)   +-------------+
     |     End     |                   |     End     |
     |   System    |                   |   System    |
     | --------    |                   | --------    |
     | | NIC |     |                   | | NIC |     |
     +-----+-------+                   +-------+-----+
           |            ____    ____          |
           |         __/    \____/    \____        |
           |        /                      \       |
        +--------|    DetNet               |----+
                  \          Networking    /
                   \   ___    __     __/
                    \__/   \___/  \____/
```
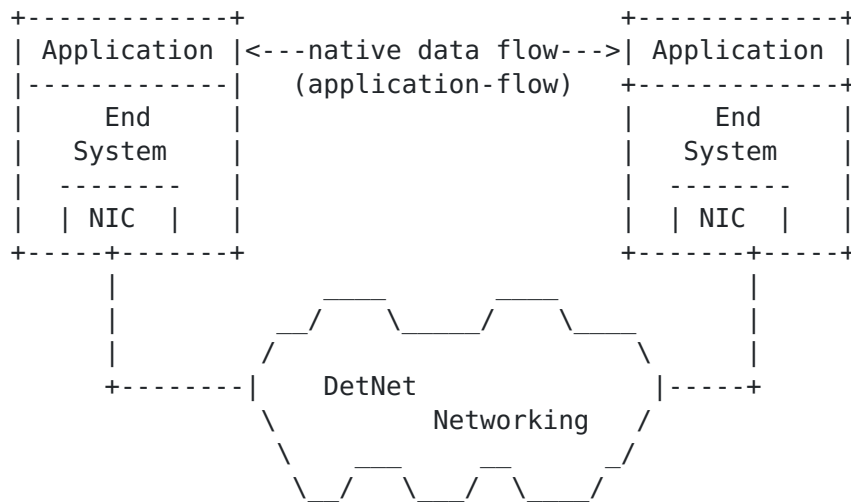
                Figure 1: End-systems connected to DetNet

End-system(s) may or may not be directly connected to the DetNet
transport network. End-systems may or may not contain DetNet
specific functionalities and are referred as "DetNet aware End-sytem"
or "DetNet unaware End-system" respectively [draft-data-plane].
(Note: [draft-data-plane] refers to "DetNet unaware end-systems" as
"TSN End-system")

o   "DetNet aware End-system" has the same forwarding paradigm as the
    DetNet transport network and it creates the DetNet-flow from the
    app-flow. DetNet aware End-system is connected via "DetNet AC" to
    the DetNet transport network.

o   "DetNet unaware End-system" originates a native data flow (app-
    flow) from which an Edge node creates a DetNet-flow (with proper
    Flow-ID and Seq-num attributes) by encapsulating native data flow
    according to the forwarding paradigm of the transport network.
    DetNet unaware End-system is connected via "Native AC" to the
    DetNet transport network.

These end-systems are shown in Figure 2

```
DetNet                         DetNet
unaware                        aware
end-system                     end-system

  _                              _
 / \                           / \
/App\ <-----App-flow--        /App\ <-----App-flow--
/--O--\    <--DetNet-flow--  /--X--\ <--DetNet-flow--
| NIC |                      | NIC |
+--+--+     +----+           +--+--+     +----+
   |     |  |T-PE|              |     |  |S-PE|
   +--------U    +-             +--------+    +-
        |  |    |                    |  |    |
        |  +----+                    |  +----+
   Native  Edge                  DetNet   Relay
      AC   Node                     AC    Node
```
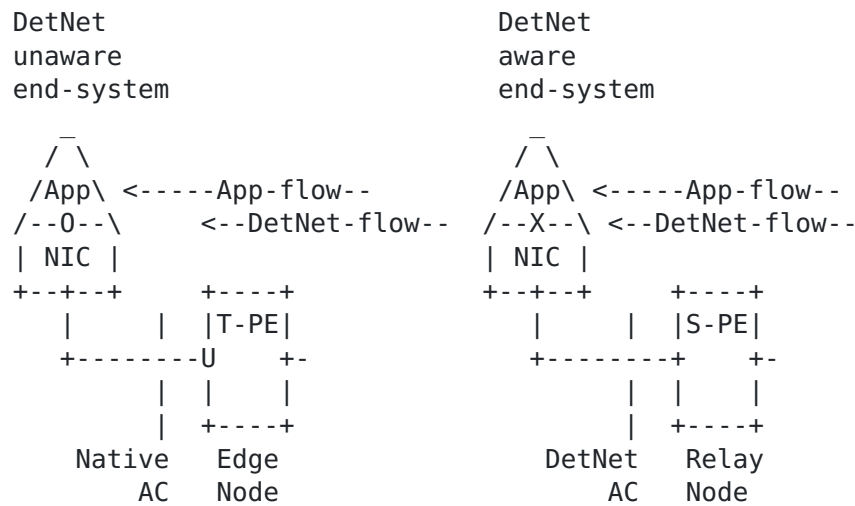
Figure 2: DetNet aware/unaware End-systems

5. DetNet service model

5.1. Service parameters

The DetNet service can be defined as a service, that provides a
capability to carry specified data flow, whether unicast or
multicast, for an application with constrained requirements towards
network performance, e.g. low packet loss rate and/or latency.

Delay and loss parameters are somewhat correlated as too late arrival
of a packet can be treated as lost. However not all applications
require hard limits on both parameters (delay and loss). For
example, some real-time applications allow graceful degradation if
loss happens (e.g., samples based processing, media distribution).
Some others may require high bandwidth connections that makes the
usage of techniques like flow duplication economically challenging or
even impossible. Some applications may not tolerate loss, but are
not delay sensitive (e.g., bufferless sensors).

Primary transport service attributes for DetNet transport are:

o Bandwidth parameter(s),

o Delay parameter(s),

o Loss parameter(s).

Time/Loss sensitive applications may have somewhat special
requirements especially for loss (e.g. no loss in two consecutives
communication cycles; very low outage time, etc.).

5.2. Service overview

The figure below shows the DetNet service related reference points
and components (Figure 3).

```
  DetNet                                                        DetNet
  unaware                                                       aware
end-system                                                    end-system

   _                                                              _
  / \                                                            / \
 /App\       +----DetNet-UNI (U)         DetNet-UNI (U)         /App\
 /--O--\      |                          [DetNet-NNI (N)]      /--X--\
 | NIC |      v                                |              | NIC |
 +--+--+          _____                 +-----+--+--------+   +--+--+
    |         /         \  _/          \    |        |        |     |
    |        /     \__/                 \   v        |        |     |
    |       /  +----+     +----+         _____     |        |     |
    |   |  /   |    |     |    |  |              \    v    _   v     |
    +--------U PE +----+ P  +----+             \   ___/ \        |
    |   |   |    |    |    |    |    |          |  /    \_   | |
    |   |   +--+-+    +----+    |        +----+ |    /       \ | |
    |    \      |                |        |    | | |   /        \ | |
    |     \     |   +----+    +--+-+   +--+ PE U-N-----N-U       U------+
    |      \    |   |    |    |    |   |  | | |    | |      \_    _/   |
    |       \   +---+ P  +----+ P  +--+  +----+ |      \___/      |
   Native    \___ |    |    |    |          /         DetNet
     AC          \  +----+__  +----+    Domain-1     Domain-2   AC
    |             \_____/  _____/                          |
    |                                                            |
    |        |       End-to-End-Service         |        |     |  |
  <------------------------------------------------------------------->
    |        |      DetNet-Service              |        |     |  |
    |        <-------------------------------->     <--------->    |
    |        <--------------------------------N---------N--------->  |
    |        |                                  |        |     |  |
    | DetLink|                                  |        |     |  |
  <-------->                               <--------->    <------>
    |        |       DetNetwork                 |        |     |  |
    |        <------------------------------->     <--------->    |
    |        <----------------------------------------------------->  |
    |        |                                  |        |     |  |
```

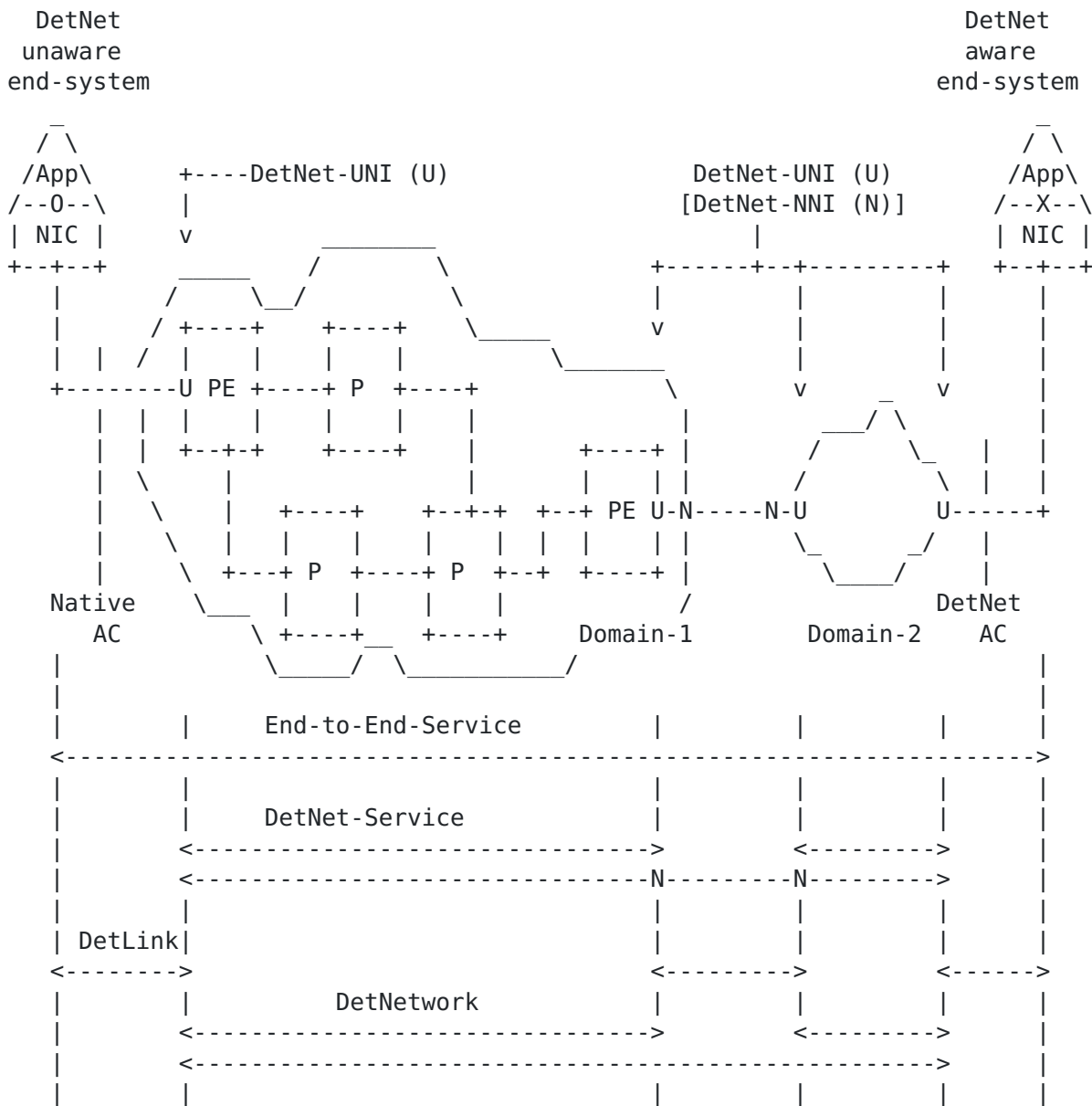   Figure 3: DetNet Service Reference Model

5.3. Reference Points

   From service model design perspective a fundamental question is the
   location of the service endpoints, i.e., where the service starts or
   ends. Two reference points can be distinguished for all DetNet use-
   cases:

   o App-flow endpoints: end-system's internal reference point.

   o DetNet-UNI: edge node UNI interface of a domain.

   App-flow endpoints (depicted as "O" and "X" on Figure 3) is a more
   challenging point from control perspective as it is an internal
   reference point. It is providing access to deterministic transport
   for the native data flow (app-flow).

   A DetNet-UNI (depicted as "U" on Figure 3) is assumed in this
   document to be a packet based reference point and provides
   connectivity over the packet network. A DetNet-UNI may add
   networking technology specific encapsulation to the app-flow and
   transport it as a DetNet-Flow over the network. There are many
   similarities regarding the functions of an app-flow endpoint ("X") of
   an DetNet aware endsystem and DetNet-UNI but there may be some
   differences. For example in-order delivery is expected over the app-
   flow endpoints ("O/X"), whereas it is considered as optional over the
   DetNet-UNI.

5.4. Service scenarios

   Using the above defined reference points two major service scenarios
   can be created:

   o End-to-End-Service: the service reaches out to final source/sink
      nodes, so it is an e2e service between application hosting devices
      (end-systems).

   o DetNet-Service: the service connects networking islands, so it is
      a service between the borders of network domain(s).

   End-to-End-Service is defined between app-flow endpoints, whereas
   DetNet-Service between DetNet-UNIs. That allows peering of same
   layers/functions.

5.5. Data flows

   For unambiguous references two detnet data flows are distinguished:

   o App-flow: data flow requiring deterministic transport between two
      app-flow endpoints, data format is application specific (e.g., bit
      stream, directly mapped in Ethernet frames, etc.).

   o DetNet-flow: data flow requiring deterministic transport between
      two DetNet-UNIs. Data format may be chaged at the DetNet-UNI to
      allow simple flow recognition/transport/etc. during forwarding
      between DetNet-UNIs (e.g., on Edge Nodes by adding further
      encapsulation to the App-flow including new domain specific Flow-
      ID and Seq-num attributes) .

[Note: In some network scenarios App-flow and DetNet-flow format
might be identical e.g., if the end-system provides an encapsulation,
that contains all information needed by detnet functionalities (e.g.,
RTP based App-flow trasported over a native IP network). In other
scenarios their encapsulation format might differ significantly e.g.,
CPRI IQ data mapped directly to Ethernet frames which have to be
transported over an MPLS based PSN.]

5.6. Service components/segments

As a reference to service components/segments the follwoing building
blocks are used:

o DetLink: direct link between two entities (node/end-system) used
   for deterministic transport.

o DetNetwork: network between DetNet-UNIs

Using DetLink and DetNetwork component/segments any detnet service
scenario can be described. For example the service between the App-
flow endpoints on Figure 3 can be composed as a DetLink-1 (between
end-system on the left and the edge node of domain-1) + DetNetwork-1
(of domain-1) + DetLink-2 (between domain-1 and domain-2) +
DetNetwork-2 (of domain-2) + DetLink-3 (between edge node of domain-2
and end-system on the right).

6. DetNet service instances

6.1. Local attributes used by DetNet functions

The three DetNet functions (Bandwidth reservation and enforcement,
Explicit routes, Packet loss protection) require two data flow
related attributes to work properly:

o Flow-ID and

o Sequence number (Seq-Num).

These attributes are local to DetNet nodes and are extracted from the
ingress packets of the node [draft-arch]. Flow-ID is used by all the
three DetNet functions, but sequence number is used only by the
duplicate elimination functionality.

Flow-ID must be unique per network domain. Its encoding format is
specific to the forwarding paradigm of the domain and to the
capabilities of intermediate nodes to identify data flows. For
example in case of "PW over MPLS" one option is to construct the
Flow-ID by the PW label and the LSP label (denoted as [PW-label;LSP-

label]). In such a case intermediate P nodes have to check all
labels to identify a DetNet-flow. An other possible option is to use
a dedicated LSP per data flow so the LSP label itself can be used as
a Flow-ID (denoted as [LSP-label]). In such a case the intermediate
P nodes do not have to check the whole label stack to recognize a
data flow (DetNet-flow).

6.2. Service instance for DetNet data flows

The DetNet network reference model is shown in Figure 4 for a DetNet-
Service scenario (i.e. between two DetNet-UNIs). In this figure the
end-systems ("A" and "B") are connected directly to the edge nodes of
the PSN ("PE1" and "PE2"). The data flow specific attachment
circuits ("AC-A" and "AC-B") are terminated in the flow specific
service instance ("SI-1" and "SI-2"). A PSN tunnel is used to
provide connectivity between the service instances. The
encapsulations used over the PSN tunnel are described in [draft-data-
plane].

This PSN tunnel is used to transport exclusivelly the DetNet data
flow packets between "SI-1" and "SI-2". The service instances are
configured to implement a flow specific routing or bridging function
depending on what connectivity (L2 or L3) the participating end-
systems require. The service instance and the PSN tunnel may or may
not be shared by multiple DetNet data flows. Sharing the service
istance by multiple DetNet-flows requires properly populated
forwarding tables of the service instance.

Serving regular traffic and DetNet data flows by the same service
instance is out-of-scope in this draft, but some related thoughts are
described in the annex.

```
     AC-A                                            AC-B
    <----->    <-------- PSN tunnel -------->     <----->

       +---------+          ___  _          +---------+
       |  +----+ |         /    \/ \_       | +----+  |
"A" ------+    | |        /        \         | |    +------ "B"
       |  |    +=========+   PSN   +========+    |  |
       |  |SI-1| |         \__      _/       | |SI-2|  |
       |  +----+ |          \___/            | +----+  |
       |PE1      |                           |     PE2|
       +---------+                           +---------+
```
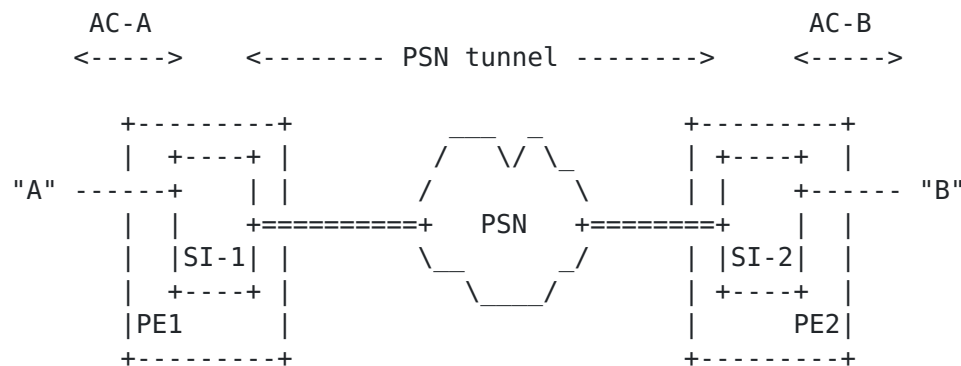
                  Figure 4: DetNet network reference model

[Note: There are differences in the usage of a "packet PW" for DetNet
traffic compared to the network model described in [rfc6658]. In the
DetNet scenario the packet PW is used exclusivelly by the DetNet data
flows, whereas RFC6658 states: "The packet PW appears as a single
point-to-point link to the client layer. Network-layer adjacency
formation and maintenance between the client equipments will follow
the normal practice needed to support the required relationship in
the client layer ... This packet pseudowire is used to transport all
of the required layer 2 and layer 3 protocols between LSR1 and
LSR2".]

7. DetNet data flows over multiple technology domains

7.1. Flow attribute mappings between layers

Transport of DetNet data flows over multiple technology domains may
require that e.g., lower layers are aware of specific flows at higher
layers. Such an "exporting of flow identification" [see draft-arch
section 4.7] is needed each time when the forwarding paradigm is
changed on the transport path (e.g., two LSRs are interconnected by a
L2 bridged domain, etc.). The three main forwarding methods
considered for deterministic networking are:

o IP routing

o MPLS label switching

o Ethernet bridging

The simplest solution for generalized flow identification could be to
define a unique Flow-ID triplet per DetNet data flow (e.g., [IP:
"IPv6-flow-label"+"IPv6-address"; MPLS: "PW-label"+"LSP-label";
Ethernet: "VLAN-ID"+"MAC-address"). This triplet can be used by the
DetNet encoding function of technology border nodes (where forwarding
paradigm changes) to adapt to capabilities of the next hop node.
They push a further (forwarding paradigm specific) Flow-ID to packets
ensuring that flows can be easily recognized by domain internal
nodes. This additional Flow-ID might be removed when the packet
leaves a given technology domain.

[Note: Seq-num attribute may require a similar functionality at
technology border nodes.]

The additional (domain specific) Flow-ID can be

o created by a domain specific function or

o derived from the original Flow-ID of the app-flow

so, that it must be unique inside the given domain. Please note,
that the original Flow-ID of the app-flow is still present in the
packet, but transport nodes may lack the function to recognize it,
that's why the additional Flow-ID is added (pushed).

7.2. Flow-ID mappings examples

IP nodes and MPLS nodes are assumed to be configured to push such an
additional (domain specific) Flow-ID when sending traffic to an
Ethernet switch (as shown in the examples below).

Figure 5 below shows a scenario where an IP end-system ("IP-A") is
connected via two Ethernet switches ("ETH-n") to an IP router ("IP-
1").

```
                             IP domain
                 <-------------------------------------------------

            +======+                                    +======+
            |L3-ID |                                    |L3-ID |
            +======+  /\                        +-----+ +======+
                     /  \       Forwards as     |     |
                    /IP-A\      per ETH-ID      |IP-1 |      Recognize
    Push  ------>  +-+----+         |           +---+-+  <----- ETH-ID
    ETH-ID        |           +----+-----+          |
                  |           v          v          |
                  |        +-----+    +-----+        |
                 +------+      |    |      +---------+
            +......+         |ETH-1+----+ETH-2|         +======+
            .L3-ID .         +-----+    +-----+         |L3-ID |
            +======+           +......+                 +======+
            |ETH-ID|           .L3-ID .                 |ETH-ID|
            +======+           +======+                 +------+
                               |ETH-ID|
                               +======+

                          Ethernet domain
                          <---------------->
```

                Figure 5: IP nodes interconnected by an Ethernet domain

"IP-A" uses the original App-flow specific ID ("L3-ID"), but as it is
connected to an Ethernet domain it has to push also an Ethernet-
domain specific flow-ID ("VLAN+multicast-MAC", referred as "ETH-ID")
before sending the packet to "ETH-1". The ethernet switch "ETH-1"
can recognize the data flow based on the "ETH-ID" and it does
forwarding towards "ETH-2". "ETH-2" switches the packet towards the

IP router. "IP-1" must be configured to receive the Ethernet Flow-ID
specific multicast stream, but (as it is an L3 node) it decodes the
data flow ID based on the "L3-ID" fields of the received packet.

Figure 6 below shows a scenario where an MPLS domain nodes ("PE-n"
and "P-m") are connected via two Ethernet switches ("ETH-n").

```
                                MPLS domain
                 <------------------------------------------------->

       +=======+                               +=======+
       |MPLS-ID|                               |MPLS-ID|
       +=======+  +-----+                +-----+ +=======+ +-----+
               |  |     |  Forwards as    |     | |         |     |
               |  |PE-1 |   per ETH-ID    | P-2 +-----------+ PE-2|
    Push  -----> +-+---+                  +---+-+            +-----+
    ETH-ID         |       +-----+----+       |  \ Recognize
                   |       v        v         |   +-- ETH-ID
                   |    +-----+    +-----+     |
                  +---+    |        |    +----+
        +.......+  |ETH-1+----+ETH-2|   +=======+
        .MPLS-ID.  +-----+    +-----+   |MPLS-ID|
        +=======+                       +=======+
        |ETH-ID |       +.......+       |ETH-ID |
        +=======+       .MPLS-ID.       +-------+
                        +=======+
                        |ETH-ID |
                        +=======+
                     Ethernet domain
                    <---------------->
```

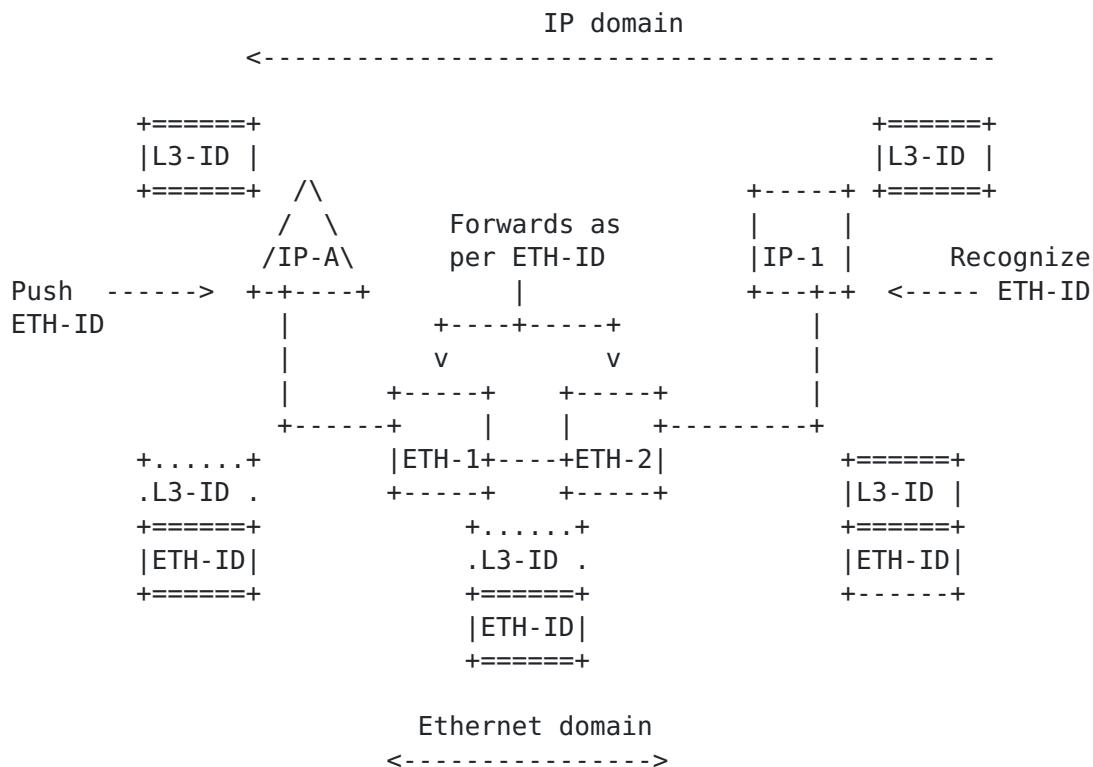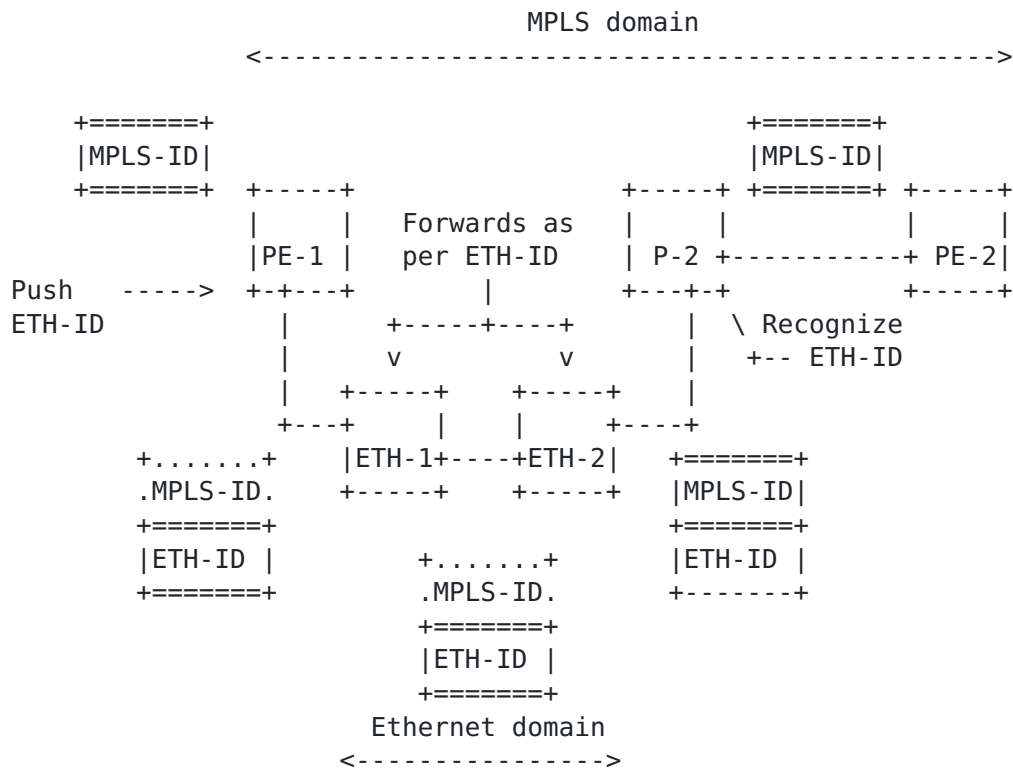   Figure 6: MPLS nodes interconnected by an Ethernet domain

   "PE-1" uses the MPLS specific ID ("MPLS-ID"), but as it is connected
   to an Ethernet domain it has to push also an Ethernet-domain specific
   flow-ID ("VLAN+multicast-MAC", referred as "ETH-ID") before sending
   the packet to "ETH-1". The ethernet switch "ETH-1" can recognize the
   data flow based on the "ETH-ID" and it does forwarding towards "ETH-
   2". "ETH-2" switches the packet towards the MPLS node ("P-2").
   "P-2" must be configured to receive the Ethernet Flow-ID specific
   multicast stream, but (as it is an MPLS node) it decodes the data
   flow ID based on the "MPLS-ID" fields of the received packet.

8. Summary

This document specifies a DetNet service model via related SAPs,
Components/Segments and Terminology .

9. IANA Considerations

N/A.

10. Security Considerations

N/A.

11. Acknowledgements

The authors wish to thank Lou Berger, Norman Finn, Jouni Korhonen and
the members of the data plane design team for their various
contributions, comments and suggestions regarding this work.

12. Annex

This Annex contains some thoughts about scenarios where the service
instance is shared by DetNet and regular traffic.

12.1. L2 service instance shared by regular and DetNet traffic

In case of a L2 VPN transport the service instance implements
bridging. In MPLS based PSN there is a full mesh of PWs between
service instances of PE nodes. Adding DetNet data flows to the
network results in a somewhat modified PW structure, as a DetNet data
flow requires its unique Flow-ID to be encoded in the packet.

```
                            +---------+
                            |     PE2|
                            | +----+  |
                 PW-12      | |SI-2|  |
            +---------------+     |   |
        +-----|---+         | +-+--+  |
        | +--+-+ |          +---|-----+
    "A" ------+    | |          |
        | |SI-1| |             |
        |  +-+-++ |            | PW-23
        |PE1 . |  |            |
        +----.-|- +            |
           . |         + --|-----+
           . |   PW-13  | +-+--+  |
           . +---------------+    |   |
           .              | |   +------ "B"
        +.................+SI-3|  |
             PW-AB        | +----+  |
                          |     PE3|
                          + --------+
```
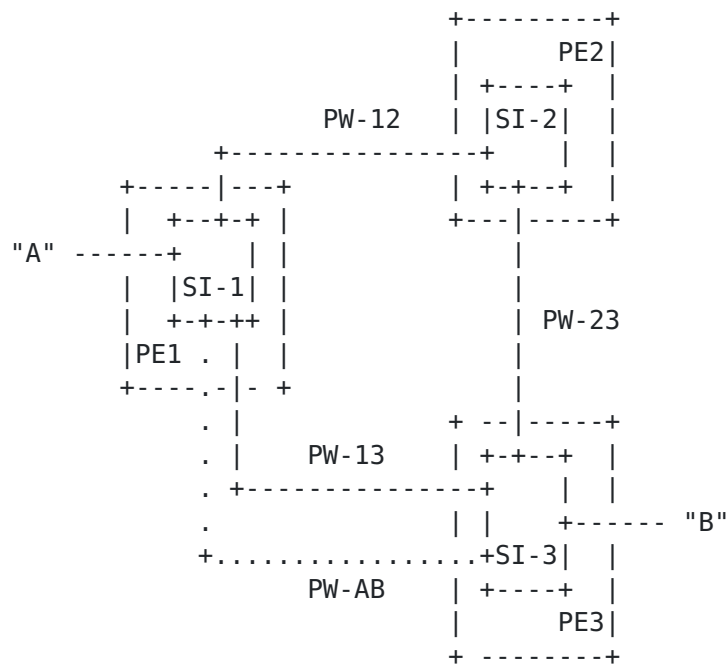
Figure 7: DetNet L2 VPN Service

Figure 7 shows a scenario where there is a DetNet data flow between
the end-systems ("A" and "B"). "SI-n" denotes the L2 VPN service
instance of "PEn". Regular traffic of the L2 VPN instance use "PW-
12", "PW-13" and "PW-23". However for transport of DetNet traffic
between "A" and "B" a separate PW ("PW-AB") have to be used. "PW-AB"
is a somewhat special PW (called here "virtual PW") and it is treated
differently than PWs used by regular traffic (i.e. PW-13, PW-12 and
PW-23). Namely, "PW-AB" is used exclusivelly by the DetNet data flow
between "A" and "B". "PW-AB" does not participate in flooding and no
MAC addresses are associated with it (not considered for the MAC
learning process). "PW-AB" may use the same LSP as "PW-13" or a
dedicated one.

Regular traffic between "A" and "B" has an encapsulation [PW-13_label
; LSP_label], whereas DetNet data flow has [PW-AB_label ; LSP_label].

12.2. L3 service instance shared by regular and DetNet traffic

In case of a L3 DetNet service the service instance implements
routing. In MPLS based PSN such a "routing service" can be provided
by IP VPNs (rfc4364). However the IP VPN service add only a single
label (VPN label) during forwarding, therefore the label stack does
not contain a "control word" (i.e., there is no field to encode a

sequence number). So, transport of DetNet data flows requires the
combination of IP VPN and PW technologies.

Adding DetNet data flows to the network results in a somewhat
modified label stack structure, as a DetNet data flow requires its
packet PW encapsulation (rfc6658).

```
                              +---------+
                              |      PE2|
                              | +----+  |
                    VPN-12    | |SI-2|  |
              +---------------+    |  |
         +-----|---+          | +-+--+  |
         |  +--+-+ |          +---|-----+
    "A" ------+    | |                  |
         |  |SI-1| |                    |
         |  +-+-++ |             | VPN-23
         |PE1 . |  |             |
         +-----.-|- +            |
              . |            + --|-----+
              . |   VPN-13   | +-+--+  |
              . +-------------+    |  |
              .              | |   +------ "B"
         +.................+SI-3|  |
              PW-AB         | +----+  |
                            |      PE3|
                            + -------+
```
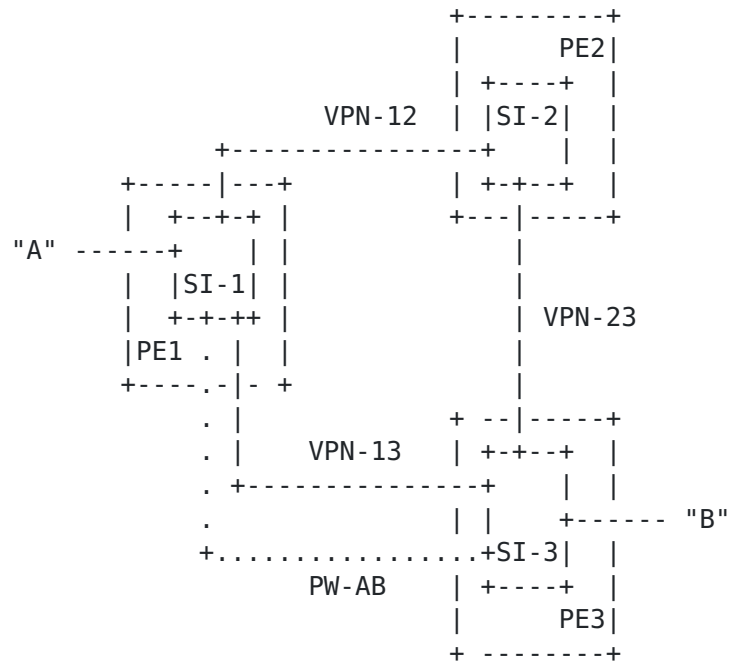
                  Figure 8: DetNet L3 VPN Service

Figure 8 shows a scenario where there is a DetNet data flow between
the end-systems ("A" and "B"). "SI-n" denotes the L3 VPN service
instance of "PEn". Regular traffic of the L3 VPN instance use as
service label "VPN-12", "VPN-13" and "VPN-23". However for transport
of DetNet traffic between "A" and "B" a PW ("PW-AB") have to be used,
what ensures that DetNet data flow can be recognized by intermediate
P nodes and a control world can be also present. "PW-AB" is used
exclusivelly by the DetNet data flow between "A" and "B". "PW-AB"
may use the same LSP as regular traffic (labeled by "VPN-13") or a
dedicated one.

Regular traffic between "A" and "B" has an encapsulation [VPN-
13_label ; LSP_label], whereas DetNet data flow has [PW-AB_label ;
LSP_label].

13. References

13.1. Normative References

[draft-arch]
          IETF, "Deterministic Networking Architecture", 2016,
          <https://datatracker.ietf.org/doc/draft-finn-detnet-
          architecture/>.

[draft-data-plane]
          IETF, "DetNet Data Plane Protocol and Solution
          Alternatives", 2016, <https://datatracker.ietf.org/doc/
          draft-dt-detnet-dp-alt/>.

[I-D.ietf-detnet-use-cases]
          Grossman, E., Gunther, C., Thubert, P., Wetterwald, P.,
          Raymond, J., Korhonen, J., Kaneko, Y., Das, S., Zha, Y.,
          Varga, B., Farkas, J., Goetz, F., and J. Schmitt,
          "Deterministic Networking Use Cases", draft-ietf-detnet-
          use-cases-09 (work in progress), March 2016.

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

13.2. Informative References

[IETFDetNet]
          IETF, "Charter for IETF DetNet Working Group", 2015,
          <https://datatracker.ietf.org/wg/detnet/charter/>.

Authors' Addresses

   Balazs Varga (editor)
   Ericsson
   Konyves Kalman krt. 11/B
   Budapest 1097
   Hungary

   Email: balazs.a.varga@ericsson.com

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

   Email: janos.farkas@ericsson.com