A YANG Data Model for In-Situ OAM

draft-ietf-ippm-ioam-yang-00
Huawei: Tianran Zhou, Jim Guichard
Cisco: Frank Brockners, Srihari Raghavan
Major Changes

• Thanks Dhruv, Reshad, Mickey for the valuable comments during the adoption call.

• Add an operational container "ioam-info" for assistant data:
  – Timestamp may not need
  – Next: buffer occupancy, and more

• Associate ACE instead of ACL with each profile.
  – The ordered list only extends within each ACL
  – ACL and IOAM are two independent module
  – Expect more kind of filters

• Remove the “action-transit”.

• Interface filter is already supported.
Overview

• Profiles
  – The IOAM model is organized as list of profiles.
  – Each profile associates with one flow and the corresponding IOAM information.
  – Multiple IOAM data types can be encapsulated into the same IOAM header.
Preallocated Tracing Profile

- The preallocated tracing option will create pre-allocated space for each node to populate its information.

```
+-rw preallocated-tracing-profile {preallocated-trace}?
 |  +--rw enabled?     boolean
 |  +--rw node-action? ioam-node-action
 |  +--rw trace-types
 |     |  +--rw use-namespace? ioam-namespace
 |     |  +--rw trace-type* ioam-trace-type
 |  +--rw enable-loopback-mode? boolean
 |  +--rw enable-active-mode? boolean
```
Incremental Tracing Profile

- The incremental tracing option contains a variable node data fields where each node allocates and pushes its node data immediately following the option header.

```
+++rw incremental-tracing-profile {incremental-trace}?
  +++rw enabled?        boolean
  +++rw node-action?    ioam-node-action
  +++rw trace-types
    | +++rw use-namespace? ioam-namespace
    | +++rw trace-type*   ioam-trace-type
  +++rw enable-loopback-mode? boolean
  +++rw enable-active-mode? boolean
  +++rw max-length?     uint32
```
Direct Export Profile

- The direct export option is used as a trigger for IOAM nodes to export IOAM data to a receiving entity (or entities).

```
   +--rw direct-export-profile {direct-export}?
       |  +--rw enabled?    boolean
       |  +--rw node-action? ioam-node-action
       |  +--rw trace-types
       |      |  +--rw use-namespace? ioam-namespace
       |      |  +--rw trace-type*  ioam-trace-type
       |  +--rw enable-loopback-mode? boolean
       +--rw enable-active-mode? boolean
    +--rw flow-id?    uint32
```
Proof of Transit Profile

• The IOAM Proof of Transit data is to support the path or service function chain verification use cases.

• It’s imported from “I-D.ietf-sfc-proof-of-transit”
Edge to Edge Profile

- The IOAM edge to edge option is to carry data that is added by the IOAM encapsulating node and interpreted by IOAM decapsulating node.

```
+---rw e2e-profile {edge-to-edge}?
    +---rw enabled?  boolean
    +---rw node-action?  ioam-node-action
    +---rw e2e-types
        +---rw use-namespace?  ioam-namespace
        +---rw e2e-type*  ioam-e2e-type
```
Next

• Comments?
• Need input for the “ioam-info”.
• Add examples on the YANG model usage.
Thank You