

# A YANG Data Model for In-Situ OAM

draft-zhou-ippm-ioam-yang-08

Huawei: Tianran Zhou, Jim Guichard

Cisco: Frank Brockners, Srihari Raghavan

# History

- Initial version:
  - March 2018
- Presented:
  - IETF 101 and 102
- Comments received from
  - Greg Mirsky, Reshad Rahman, Tom Petch
- Several revisions kept aligning with the latest IOAM data draft and IOAM-DEX draft
- The latest version corrected the admin type things
  - References, Copyright, Tree diagrams

# 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.

```
module: ietf-ioam
  +--rw ioam
    +--rw ioam-profiles
      +--rw admin-config
        | +--rw enabled?   boolean
      +--rw ioam-profile* [profile-name]
        +--rw profile-name          string
        +--rw filter
          | +--rw filter-type?   ioam-filter-type
          | +--rw acl-name?      -> /acl:acls/acl/name
        +--rw protocol-type?        ioam-protocol-type
        +--rw incremental-tracing-profile {incremental-trace}?
          | ...
        +--rw preallocated-tracing-profile {preallocated-trace}?
          | ...
        +--rw direct-export-profile {direct-export}?
          | ...
        +--rw pot-profile {proof-of-transit}?
          | ...
        +--rw e2e-profile {edge-to-edge}?
          | ...
          ...
```

# 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"

```
+---rw pot-profile {proof-of-transit}?
  +---rw enabled?                boolean
  +---rw active-profile-index?    pot:profile-index-range
  +---rw pot-profile-list* [pot-profile-index]
    +---rw pot-profile-index      profile-index-range
    +---rw prime-number           uint64
    +---rw secret-share           uint64
    +---rw public-polynomial      uint64
    +---rw lpc                    uint64
    +---rw validator?            boolean
    +---rw validator-key?        uint64
    +---rw bitmask?              uint64
      +---rw opot-masks
    +---rw downstream-mask*      uint64
    +---rw upstream-mask*       uint64
```

# 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?
- How about adopting this draft as the starting point for IOAM configurations?

Thank You