OPSAWG L. Melegassi
Internet-Draft Catellix
Intended status: Standards Track 28 May 2026
Expires: 29 November 2026

A YANG Data Model for Multi-Vantage Path Snapshots (MVPS)

draft-melegassi-opsawg-mvps-yang-model-00

Abstract

This document defines a YANG data model for Multi-Vantage Path Snapshots (MVPS): vendor-neutral, multi-vantage enriched traceroute observations whose reporting model is aligned with RFC 9198 (Advanced Unidirectional Route Assessment). The model is the normative publication of the MVPS bundle as a YANG module and is the subtree that the MVPS telemetry-export specification subscribes to over YANG-Push.

The module is CORE-neutral: it carries measurement facts only. It makes no performance, scoring, or detection claim. All properties stated in this document are structural and are backed by a machine-checkable receipt.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Multi-Vantage Path Snapshots (MVPS) collect enriched traceroute observations from several vantages and bind them into a single canonical bundle. The bundle format, its JSON-Schema sibling, and the coherence detection mathematics are specified elsewhere in the MVPS family. A telemetry-export specification additionally maps MVPS observations onto standard carriers, including YANG-Push [RFC8641].

That export mapping presumes a published YANG subtree to subscribe to. This document supplies it: it publishes the MVPS YANG module normatively, defines its instance-identifier structure, and states the structural properties on which interoperable configuration, retrieval (NETCONF/RESTCONF), and subscription (YANG-Push) depend.

This is a data-model document. It deliberately makes NO performance or detection-latency claim. Every property in Section 5 is structural: a deterministic fact about the module text or about any conformant instance, verifiable by the companion receipt and independent of any measurement.

The module models measurement facts only (CORE neutrality). Any analytic verdict, score, or machine-learning output is OUT OF SCOPE for this module and MUST be carried in the namespaced extension slot defined by the MVPS extension mechanism.

2. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals.

Vantage: one observation origin (an active server or an edge network element) that contributes a Member Route or a consolidated Route Ensemble, per [RFC9198] Section 4.1.

Hop: a single hop singleton h(i,j) along the observed path, per

[RFC9198] Section 3.4.

Bundle: the top-level MVPS container for one snapshot, encoded in JSON per [RFC7951].

CORE-neutral: carrying measurement facts only, with no analytic verdict, score, or inference.

3. Design Principles

P1 CORE neutrality. The module carries only measurement facts. No analytic verdict, score, or AI/ML output is part of this canonical model.

P2 Externalized vendor signals. Vendor-specific or analytic signals MUST live outside this module, under reverse-DNS namespaced keys in an extension slot. Consumers MUST tolerate unknown keys (the spirit of [RFC6648]).

P3 Reproducible fingerprints. Each vantage carries three path fingerprints that are deterministic functions of its hop list. Recomputation reproduces them exactly, so any silent edit is detectable.

P4 Standards alignment. Per-hop fields materialise the AURA Hop singleton ([RFC9198] Section 3.4) with optional ICMP interface identifiers ([RFC5837]) and Round-Trip Delay quartiles computed via the P^2 algorithm referenced by [RFC9198].

P5 Incremental implementability. The top-level node is a presence container and carries no mandatory child leaf, per [RFC8407] Section 4.10.

4. Model Overview (Tree Diagram)

The following tree diagram uses the notation of [RFC8340].


```
module: catellix-mvps
  +--rw mvps!
     +--rw mvps-schema?                string
     +--rw mvps-version?               string
     +--rw catellix-platform-release?  string
     +--rw document-generated-at?      yang:date-and-time
     +--rw destination?                string
     +--rw vantage-count?              uint32
     +--rw vantages* [origin-label]
        +--rw vantage-role            identityref
        +--rw origin-label            string
        +--rw observed-at?            yang:date-and-time
        +--rw path-fingerprints
        |  +--rw path-fp-ip-chain-sha256-trunc128     sha256-hex
        |  +--rw path-fp-as-path-sha256-trunc64        sha256-hex
        |  +--rw path-fp-country-path-sha256-trunc64   sha256-hex
        +--rw as-path-inferred*       union
        +--rw country-path-inferred*  string
        +--rw hop-count?              uint8
        +--rw hops* [hop-number]
```

```
            +--rw hop-number            uint8
            +--rw ip-reported           inet:ip-address
            +--rw rtt-reported?         string
            +--rw rpki-origin-validation?  rpki-validation-state
            +--rw routing-snapshot
            +--rw rtd-quartiles
            +--rw mpls-labels* [label]
            +--rw rtt-samples-ms*       decimal64
            +--rw geo-hint!
```

The full set of leaves is defined by the module in Section 7.

5. Structural Properties

The properties below are proven, not asserted. Each maps to a check in the companion validator (scripts/validate_yang_model.py, 8/8 PASS) whose result is recorded in the receipt (evidence/yang_model_receipt.json).

T-YANG-WF (Well-formedness): the module is YANG 1.1 with a single namespace, a rooted presence container "mvps", keyed lists "vantages" (key origin-label) and "hops" (key hop-number) each with min-elements 1, ordered-by user collections, and mandatory list keys.

T-YANG-8407 (RFC 8407 Section 4.10): the top-level node is a presence container and has no mandatory child leaf, so the module can be implemented incrementally.

T-YANG-RT (Round-trip losslessness): for any conformant instance I, decode(encode(I)) = I under [RFC7951], and the order of every ordered-by user collection is preserved.

T-YANG-FP (Fingerprint determinism): the three path fingerprints are deterministic functions of the modeled fields; recomputation reproduces the stored values exactly, and the canonical JSON ([RFC8785]) of the encoding is stable. This carries the bundle’s tamper-evidence property into the model.

T-YANG-SENT (Sentinel bijection): the AS-path union sentinel "unknown" maps to the JSON-Schema sibling token "?" by a bijection on (AS-number) union {sentinel}; no real AS number collides with the sentinel.

T-YANG-CORE (CORE neutrality): the module contains no analytic verdict/score/ML leaf; vendor signals are externalized to the extension slot; and the core detection inputs (hop-number, ip-reported, rtt-samples) are invariant to the presence or absence of optional hint containers.

T-YANG-PUSH (Addressability): the module is a single rooted subtree whose every list is fully keyed, so every node has a unique instance-identifier and a YANG-Push [RFC8641] subtree or xpath subscription onto /catellix-mvps:mvps is well-defined.

T-YANG-PARITY (Schema parity): on the load-bearing constraints (version pattern, vantage cardinality, min-elements), the YANG module and the JSON-Schema sibling agree.

6. Relationship to Other MVPS Documents

This module publishes the data model that the MVPS bundle format defines. The fingerprint method (T-YANG-FP) is the bundle's method. The CORE-neutrality and externalized-extension rule (T-YANG-CORE) are the model-level form of the MVPS extension mechanism's core-invariance property. The addressability property (T-YANG-PUSH) discharges the precondition that the MVPS telemetry-export specification assumes when it maps events onto YANG-Push.

7. The YANG Module

The normative module is "catellix-mvps", revision 2026-05-14, namespace "https://catellix.com/yang/catellix-mvps". For length, the complete module text is maintained in the source repository file schema/catellix-mvps.yang and will be inlined verbatim in the next revision of this document. Implementers MUST use the module exactly as published there; the tree diagram in Section 4 is informative.

The module imports ietf-inet-types and ietf-yang-types [RFC6991]. It defines the identities vantage-role (with derived catellix-aurix-server and edge-network-element), and the typedefs sha256-hex, latency-class, rpki-validation-state, and holder-kind.

On WG adoption, the module is expected to be renamed to an "ietf-" prefixed module under an IANA-assigned namespace; the structural properties of Section 5 are invariant to that rename.

8. IANA Considerations

This document requests that IANA register the following URI in the "ns" subregistry of the "IETF XML Registry" [RFC3688] on adoption (placeholder until the module is renamed to an ietf- module):

```
URI:  urn:ietf:params:xml:ns:yang:ietf-mvps
Registrant Contact:  The IESG.
XML:  N/A; the requested URI is a YANG module namespace.
```

This document requests that IANA register the following YANG module in the "YANG Module Names" registry [RFC6020]:

```
name:        ietf-mvps
namespace:   urn:ietf:params:xml:ns:yang:ietf-mvps
prefix:      mvps
reference:   This document
```

Until adoption, the module ships under the vendor name "catellix-mvps" and namespace "https://catellix.com/yang/catellix-mvps".

9. Security Considerations

The model is to be accessed via a secure transport with mutual authentication, for example NETCONF over SSH or RESTCONF over TLS, and YANG-Push subscriptions over the same.

The data nodes are operational measurement facts. None carries a subscriber-precise location or payload; geographic fields are coarse hints only, and flow identity is republished as an anonymous fingerprint rather than the underlying values.

Because the path fingerprints are deterministic (T-YANG-FP), a reader can detect tampering of the hop, AS, or country lists by recomputation. This model does not, by itself, provide confidentiality, integrity, or origin authentication of a bundle in transit; those are provided by the transport and by the MVPS signing/anchoring documents.

The module is CORE-neutral (T-YANG-CORE): it cannot, by construction, carry an analytic verdict that an attacker could spoof inside the canonical model. Such signals are confined to the namespaced extension slot and are out of scope here.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, July 2013.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, August 2016.

[RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, August 2016.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, March 2018.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, September 2019.

10.2. Informative References

[RFC5837] Atlas, A., Ed., Bonica, R., Ed., Pignataro, C., Ed., Shen, N., and JR. Rivers, "Extending ICMP for Interface and Next-Hop Identification", RFC 5837, April 2010.

[RFC6648] Saint-Andre, P., Crocker, D., and M. Nottingham, "Deprecating the 'X-' Prefix and Similar Constructs in Application Protocols", BCP 178, RFC 6648, June 2012.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, March 2018.

[RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, March 2018.

[RFC8407] Bierman, A., "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", BCP 216, RFC 8407, October 2018.

[RFC8785] Rundgren, A., Jordan, B., and S. Erdtman, "JSON Canonicalization Scheme (JCS)", RFC 8785, June 2020.

[RFC9198] Alvarez-Hamelin, J., Morton, A., Fabini, J., Pignataro, C., and R. Geib, "Advanced Unidirectional Route Assessment (AURA)", RFC 9198, May 2022.

Author's Address

Leonardo Melegassi
Catellix
Email: melegassi@catellix.com