NMOP
Internet-Draft
Intended status: Experimental
Expires: 23 July 2026

T. Graf
W. Du
Swisscom
A. Huang Feng
INSA-Lyon
V. Riccobene
Huawei

19 January 2026

Semantic Metadata Annotation for Network Anomaly Detection
draft-ietf-nmop-network-anomaly-semantics-05

Abstract

This document explains the motivation for defining semantic metadata annotations to help testing, validating and comparing Outlier and Symptom detection systems. These semantic annotations can be supported by supervised and semi-supervised machine learning algorithms and enable data exchange among network operators, vendors and academia, making anomalies apprehensible for humans. The proposed semantics uniforms the network anomaly data exchange between operators and vendors to improve their Service Disruption Detection Systems.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Operations and Management Area Working Group Working Group mailing list (nmop@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/nmop/.

Source for this draft and an issue tracker can be found at https://github.com/network-analytics/draft-netana-nmop-network-anomaly-semantics/.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

[I-D.ietf-nmop-network-anomaly-architecture] provides an overall introduction into how anomaly detection is applied to the IP network domain and which operational data are needed. It approaches the problem space by automating what a network engineer would normally do when verifying a network connectivity service, monitoring the different network planes to understand wherever one network plane affects another negatively.

As a Service Disruption Detection Systems may need to be fine tuned to effectively maintain good anomaly detection rates, the system need to generate analytical data that is reviewed by a network engineer. This process is defined in [I-D.ietf-nmop-network-anomaly-lifecycle], where the human engineer can be kept out of the monitoring process but needs to be involved in the alarm verification process.

This document describes what information is needed to understand the analytical results produced by the Service Disruption Detection System. The document proposes a set of semantically structured terms that can be used by a Service Disruption Detection System for comparing the results systematically, setting the baselines for supervised machine learning algorithms that require labeled operational data.

This document proposes two YANG Service Models, a service topology model in Section 4.3 to describe the topology context and a YANG symptom model in Section 4.2 to describe the symptoms defined In Section 3. Section 4.4 examples above Service Models in an Apache AVRO data model based on 'ietf-relevant-state.yang' data model defined in [I-D.ietf-nmop-network-anomaly-lifecycle].

2. Conventions and Definitions

2.1. Terminology

This document makes use of the terms defined in [I-D.ietf-nmop-network-anomaly-architecture], [I-D.ietf-nmop-terminology] and [RFC8969].

The following terms are used as defined in [I-D.ietf-nmop-network-anomaly-architecture]:

* Outlier Detection

* Contextual Outlier

* Service Disruption Detection

* Service Disruption Detection System

The following terms are used as defined in [I-D.ietf-nmop-network-anomaly-lifecycle] :

* Concern Score

The following terms are used as defined in [I-D.ietf-nmop-terminology]:

* System

* State

* Problem

* Symptom

* Alarm

The following terms are defined in [I-D.ietf-nmop-terminology] but the meaning has been adapted for the use in Outlier Detection:

* Symptom: An observable value, change, State, event, or condition which itself isn't classified as an outlier, but may be correlated to an outlier and thus is considered as an indication of a Problem or potential Problem. A Symptom may be part of a collective outlier.

The following terms are used as defined in [RFC8969] :

* Service Model

3. Observed Symptoms

Observed network Symptoms are specified and categorized according to the following scheme:

Action: The action that a network node performed for a packet in the Forwarding Plane, a path or adjacency in the Control Plane, or the representation of resource state in the Management Plane or statistical changes recorded by the resources and reported in the Management Plane. For Forwarding Plane we distinguish between *missing*, where the packet drop occurred outside the measured network node, *drop*, where the packet drop was performed by the measured network node, and *delay*, which defines the on-path delay measured on the network node. For Control Plane we distinguish between *reachability*, which refers to a change in

the routing or forwarding information base (RIB/FIB) and *adjacency* which refers to a change in a peering or link-layer resolution. For Management Plane we refer to *state* or *statistical* change on the interface.

Reason: For each action, the reason describe why this action was performed. For drops in Forwarding Plane we distinguish between *Unreachable*, because network layer reachability information was missing, *Administered*, because an administrator configured a rule preventing the forwarding of this packet, and *Corrupt*, where the network node was unable to determine the forwarding path due to a packet, software or hardware error. For on-path delay we distinguish between *Minimum*, *Average* and *Maximum* delay for a given flow. For Control Plane, we distinguish wherever a the reachability action was due to path *updates* or *withdraws* or the adjacency was *established* or *teared down*. For Management Plane, we distinguish between interfaces states that are shown as *up* and *down*, and statistical counters that refer to *errors*, packet *discards* or *unknown protocol* counters.

Trigger: For each reason, the trigger describe why a network node has chosen that action.

Table 1 consolidates the list of common symptoms related to the forwarding plane, defining the triplets action, reason and trigger.

Action	Reason	Trigger
Missing	Drop	Outside Monitored Domain
Drop	Unreachable	next-hop
Drop	Unreachable	link-layer
Drop	Unreachable	Time To Life expired
Drop	Unreachable	Fragmentation needed and Don’t Fragment set
Drop	Administered	Access-List
Drop	Administered	Unicast Reverse Path Forwarding
Drop	Administered	Discard Route
Drop	Administered	Policed
Drop	Administered	Shaped
Drop	Corrupt	Bad Packet
Drop	Corrupt	Bad Egress Interface
Delay	Min	-
Delay	Mean	-
Delay	Max	-

Table 1: Description of symptoms and their actions, reason and trigger for Forwarding Plane.

Table 2 consolidates the list of common symptoms related to control plane, describing their actions, reasons and triggers.

Action	Reason	Trigger
Reachability	Update	Imported
Reachability	Update	Received

Reachability	Withdraw	Received
Reachability	Withdraw	Peer Down
Reachability	Withdraw	Suppressed
Reachability	Withdraw	Stale
Reachability	Withdraw	Route Policy Filtered
Reachability	Withdraw	Maximum Number of Prefixes Reached
Adjacency	Established	Peer
Adjacency	Established	Link-Layer
Adjacency	Locally Teared Down	Peer
Adjacency	Remotely Teared Down	Peer
Adjacency	Locally Teared Down	Link-Layer
Adjacency	Remotely Teared Down	Link-Layer
Adjacency	Locally Teared Down	Administrative
Adjacency	Remotely Teared Down	Administrative
Adjacency	Locally Teared Down	Maximum Number of Prefixes Reached
Adjacency	Remotely Teared Down	Maximum Number of Prefixes Reached
Adjacency	Locally Teared Down	Transport Connection Failed
Adjacency	Remotely Teared Down	Transport Connection Failed

Table 2: Description of symptoms and their actions, reasons and triggers related to Control Plane.

Table 3 consolidates the list of common symptoms related to management plane, defining the triplets action, reason and trigger.

Action	Reason	Trigger
Interface State	Up	Link-Layer
Interface State	Down	Link-Layer
Interface Statistics	Errors	-
Interface Statistics	Discards	-
Interface Statistics	Unknown Protocol	-

Table 3: Description of symptoms and their actions, reasons and triggers for Management Plane.

4. Semantic Metadata

Operational Metadata adds additional context to collected metrics. For instance, in a network, the software version of the network node defines the version of the software release that generated Management Plane metrics [I-D.ietf-opsawg-collected-data-manifest]. Semantic Metadata, on the other hand, defines the meaning or ontology of the annotated data. In this section a YANG model is defined in order to provide a structure for the metadata related to anomalies occurred in a network. The module is intended to describe the metadata used for "annotating" the operational data collected from the network nodes, which include time series data, logs, as well as other forms of data that is "time-bounded". The aspects discussed in this document are grouped under the concept of "anomaly" which represents a collection of symptoms. The anomaly overall has a set of parameters that describe the overall behavior of the network in a given time-window including all the observed symptoms and outliers.

4.1. Overview of the Models for the Symptom Semantic Metadata

This section defines two YANG models, one defining a placeholder for the action reason trigger defined in this document, and one defining service topology information related to the anomaly.

4.2. YANG Module 'ietf-network-anomaly-symptom-cbl'

4.2.1. YANG Tree

Figure 1 contains the YANG tree diagram [RFC8340] of the 'ietf-network-anomaly-symptom-cbl' module. It augments the 'ietf-relevant-state' module defined in [I-D.ietf-nmop-network-anomaly-lifecycle].

For each Symptom, the following parameters can be assigned: an Action, a Reason and a Trigger describing the Symptom; a Concern Score indicating how critical the Symptom is; and the associated network plane.

Where the season enumeration declares wherever a workday or a holiday hase been taken into consideration for Contextual Outliers. The template describes which approach and parameters have been used in the Service Disruption Detection as described in Section 3.2 of [I-D.ietf-nmop-network-anomaly-architecture]

```
module: ietf-network-anomaly-symptom-cbl

  augment /rsn:relevant-state/rsn:anomaly/rsn:symptom:
    +--rw action?          string
    +--rw reason?          string
    +--rw trigger?         string
    +--rw network-plane?   enumeration
    +--rw template?        string
    +--rw season?          enumeration
  augment /rsn:relevant-state-notification/rsn:anomaly/rsn:symptom:
    +-- action?          string
    +-- reason?          string
    +-- trigger?         string
    +-- network-plane?   enumeration
    +-- template?        string
    +-- season?          enumeration
```

Figure 1: YANG tree diagram for 'ietf-network-anomaly-symptom-cbl' module.

The module augments the anomaly of the 'relevant-state' container and the 'relevant-state-notification' of 'ietf-relevant-state' module defined in [I-D.ietf-nmop-network-anomaly-lifecycle]. The 'relevant-state' container is used for modifying the Symptom data in the Postmortem system, while the 'relevant-state-notification' is used for messaging from the Alarm Aggregation to the Postmortem and the Alarm and Problem Management system.

4.2.2. YANG Module

The YANG module has a grouping defining Action, Reason and Trigger and how symptom attributes to the network planes.

<CODE BEGINS> file "ietf-network-anomaly-symptom-cbl@2025-11-15.yang"

```
module ietf-network-anomaly-symptom-cbl {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-network-anomaly-symptom-cbl";
  prefix smcblsymptom;

  import ietf-relevant-state {
    prefix rsn;
    reference
      "RFC XXX: Relevant State and Relevant State Notification";
  }

  organization
    "IETF NMOP (Network Management Operations) Working Group";
  contact
    "WG Web:   <http:/tools.ietf.org/wg/netconf/>
     WG List:  <mailto:nmop@ietf.org>

     Editor:   Thomas Graf
               <mailto:thomas.graf@swisscom.com>
               Wanting Du
               <mailto:wanting.du@swisscom.com>
               Alex Huang Feng
               <mailto:alex.huang-feng@insa-lyon.fr>
               Vincenzo Riccobene
               <mailto:vincenzo.riccobene@huawei-partners.com>";
  description
    "This module defines the semantic grouping to be used by a
     Service Disruption Detection Systems. The defined objects is
     used to augment the anomaly container. Describing the
     symptoms action and reason.

     Copyright (c) 2025 IETF Trust and the persons
     identified as authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject
     to the license terms contained in, the Revised BSD License
     set forth in Section 4.c of the IETF Trust’s Legal Provisions
     Relating to IETF Documents
     (https://trustee.ietf.org/license-info).
```

```
        All revisions of IETF and IANA published modules can be found
        at the YANG Parameters registry
        (https://www.iana.org/assignments/yang-parameters).

        This version of this YANG module is part of RFC XXXX; see
        the RFC itself for full legal notices.";

      revision 2025-11-15 {
        description
          "Initial version";
        reference
          "RFC XXX: Semantic Metadata Annotation for Network Anomaly
           Detection";
      }

      grouping cbl-symptom {
        description
          "Semantic metadata assocaited to a symptom for a detected
           connectivity service anomaly.";
        leaf action {
          type string;
          description
            "Operation performed by a network node when forwarding a
             packet.";
        }
        leaf reason {
          type string;
          description
            "Reason associated to the action performed by the network
             node.";
        }
        leaf trigger {
          type string;
          description
            "Describes what triggered the network node to this action.";
        }
        leaf network-plane {
          type enumeration {
            enum forwarding {
              description
                "Symptom associated to the Forwarding Plane.";
            }
            enum control {
              description
                "Symptom associated to the Control Plane.";
            }
            enum management {
              description
```

```
                "Symptom associated to the Management Plane.";
            }
          }
          description
            "Associated network plane.";
        }
        leaf template {
          type string;
          mandatory false;
          description
            "A group of configuration parameters contributing to the symptom
             detection computation";
          reference
            "Section 3.2 in draft-ietf-nmop-network-anomaly-architecture.";
        }
        leaf season {
          type enumeration {
            enum workday {
              description
                "Contextual outlier associated to workday.";
            }
            enum holiday {
              description
                "Contextual outlier associated to holiday.";
            }
          }
          description
            "Associated season.";
        }
      }

      augment "/rsn:relevant-state/rsn:anomaly"
            + "/rsn:symptom" {
        description
          "Provide extension for the symptom description,
           specifically for connectivity services to the
           relevant state container";
        uses cbl-symptom;
      }

      augment "/rsn:relevant-state-notification/rsn:anomaly"
            + "/rsn:symptom" {
        description
          "Provide extension for the symptom description,
           specifically for connectivity services to the
           relevant state notification";
        uses cbl-symptom;
      }
```

```
}
<CODE ENDS>
```

4.3. YANG Module 'ietf-network-anomaly-service-topology'

4.3.1. YANG Tree

The YANG module has a service and a vpn-termination grouping defining a 'vpn-id', a 'vpn-name' 'site-ids' and a 'change-id' with 'start' and 'end time' for 'service' and 'hostname', 'VRF ID', 'VRF Name', 'BGP route-distinguisher', 'BGP peer ip address', 'BGP path next-hop', 'node interface-id' and 'node interface-name' for 'node-termination' list and 'hostname', 'BGP route-distinguisher', 'BGP path next-hop' and 'BGP peer ip address' for 'network-termination' list.

Within the NMOP working group we discuss with the SIMAP authors which existing YANG nodes instead could be used to facilitate a service and network topology context view.

module: ietf-network-anomaly-service-topology

```
  augment /rsn:relevant-state/rsn:service:
    +--:(l2vpn)
    |  +--rw l2vpn-service* [vpn-id]
    |     +--rw vpn-id               string
    |     +--rw uri?                 inet:uri
    |     +--rw vpn-name?            string
    |     +--rw site-ids*            string
    |     +--rw change-id?           yang:uuid
    |     +--rw change-start-time?   yang:date-and-time
    |     +--rw change-end-time?     yang:date-and-time
    +--:(l3vpn)
       +--rw l3vpn-service* [vpn-id]
          +--rw vpn-id               string
          +--rw uri?                 inet:uri
          +--rw vpn-name?            string
          +--rw site-ids*            string
          +--rw change-id?           yang:uuid
          +--rw change-start-time?   yang:date-and-time
          +--rw change-end-time?     yang:date-and-time
  augment /rsn:relevant-state-notification/rsn:service:
    +--:(l2vpn)
    |  +-- l2vpn-service* [vpn-id]
    |     +-- vpn-id               string
    |     +-- uri?                 inet:uri
    |     +-- vpn-name?            string
    |     +-- site-ids*            string
```

```
       |     +-- change-id?           yang:uuid
       |     +-- change-start-time?   yang:date-and-time
       |     +-- change-end-time?     yang:date-and-time
       +--:(l3vpn)
          +-- l3vpn-service* [vpn-id]
             +-- vpn-id               string
             +-- uri?                 inet:uri
             +-- vpn-name?            string
             +-- site-ids*            string
             +-- change-id?           yang:uuid
             +-- change-start-time?   yang:date-and-time
             +-- change-end-time?     yang:date-and-time
     augment /rsn:relevant-state/rsn:anomaly:
       +--rw vpn-node-terminations* [hostname vrf-name]
          +--rw hostname               inet:host
          +--rw vrf-id?                uint32
          +--rw vrf-name               string
          +--rw route-distinguisher?   string
          +--rw interface-id*          uint32
          +--rw interface-name*        string
          +--rw peer-ip*               inet:ip-address
          +--rw next-hop*              inet:ip-address
     augment /rsn:relevant-state-notification/rsn:anomaly:
       +-- vpn-node-terminations* [hostname vrf-name]
          +-- hostname               inet:host
          +-- vrf-id?                uint32
          +-- vrf-name               string
          +-- route-distinguisher?   string
          +-- interface-id*          uint32
          +-- interface-name*        string
          +-- peer-ip*               inet:ip-address
          +-- next-hop*              inet:ip-address
```

4.3.2. YANG Module

The 'ietf-network-anomaly-service-topology' module defines reusable groupings for augmenting the 'relevant-state' model. It defines placeholders for defining VPN information that is associated to the relevant state.

<CODE BEGINS>

```
  file "ietf-network-anomaly-service-topology@2025-11-15.yang"
module ietf-network-anomaly-service-topology {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-network-anomaly-service-topology";
  prefix smtopology;
```

```
     import ietf-inet-types {
       prefix inet;
       reference
         "RFC 6991: Common YANG Data Types";
     }
     import ietf-yang-types {
       prefix yang;
       reference
         "RFC 6991: Common YANG Data Types";
     }
     import ietf-relevant-state {
       prefix rsn;
       reference
         "RFC XXX: An Experiment: Network Anomaly Lifecycle";
     }

     organization
       "IETF NMOP (Network Management Operations) Working Group";
     contact
       "WG Web:   <http:/tools.ietf.org/wg/netconf/>
        WG List:  <mailto:nmop@ietf.org>

        Editor:   Thomas Graf
                  <mailto:thomas.graf@swisscom.com>
                  Wanting Du
                  <mailto:wanting.du@swisscom.com>
                  Alex Huang Feng
                  <mailto:alex.huang-feng@insa-lyon.fr>
                  Vincenzo Riccobene
                  <mailto:vincenzo.riccobene@huawei-partners.com>";
     description
       "This module defines the symptom container to be used by a network
        anomaly detection system. The defined objects can be used to
        augment operational network collected observability data and
        analytical problem data equally. Describing the relevant-state
        of observed symptoms.

        Copyright (c) 2025 IETF Trust and the persons
        identified as authors of the code.  All rights reserved.

        Redistribution and use in source and binary forms, with or
        without modification, is permitted pursuant to, and subject
        to the license terms contained in, the Revised BSD License
        set forth in Section 4.c of the IETF Trust's Legal Provisions
        Relating to IETF Documents
        (https://trustee.ietf.org/license-info).

        All revisions of IETF and IANA published modules can be found
```

```
        at the YANG Parameters registry
        (https://www.iana.org/assignments/yang-parameters).

        This version of this YANG module is part of RFC XXXX; see
        the RFC itself for full legal notices.";

      revision 2025-11-15 {
        description
          "Initial version";
        reference
          "RFC XXX: Semantic Metadata Annotation for Network Anomaly Detection";
      }

      grouping l2vpn-service {
        description
          "Connectivity service of type VPN. This grouping is
           used to augment the relevant-state container.";
        list l2vpn-service {
          key "vpn-id";
          description
            "List of VPN connectivity services of interest.";
          leaf vpn-id {
            type string;
            mandatory true;
            description
              "Unique ID of the VPN connectivity service.";
          }
          leaf uri {
            type inet:uri;
            description
              "URI to viusalize the VPN connectivity service inventory.";
          }
          leaf vpn-name {
            type string;
            description
              "Name of the VPN connectivity service.";
          }
          leaf-list site-ids {
            type string;
            description
              "List of unique site ID’s of the VPN connectivity service.";
          }
          leaf change-id {
            type yang:uuid;
            description
              "Unique identifier of VPN connectivity service maintenance
               window within the relevant-state window.";
          }
```

```
        leaf change-start-time {
          type yang:date-and-time;
          description
            "Start date and time of the VPN connectivity service
             maintenance window within the relevant-state window.";
        }
        leaf change-end-time {
          type yang:date-and-time;
          description
            "End date and time of the VPN connectivity service
             maintenance window within the relevant-state window.";
        }
      }
    }

    grouping l3vpn-service {
      description
        "Connectivity service of type VPN. This grouping is
         used to augment the relevant-state container.";
      list l3vpn-service {
        key "vpn-id";
        description
          "List of VPN connectivity services of interest.";
        leaf vpn-id {
          type string;
          mandatory true;
          description
            "Unique ID of the VPN connectivity service.";
        }
        leaf uri {
          type inet:uri;
          description
            "URI to viusalize the VPN connectivity service inventory.";
        }
        leaf vpn-name {
          type string;
          description
            "Name of the VPN connectivity service.";
        }
        leaf-list site-ids {
          type string;
          description
            "List of unique site ID's of the VPN connectivity service.";
        }
        leaf change-id {
          type yang:uuid;
          description
            "Unique identifier of VPN connectivity service maintenance
```

```
            window within the relevant-state window.";
        }
        leaf change-start-time {
          type yang:date-and-time;
          description
            "Start date and time of the VPN connectivity service
             maintenance window within the relevant-state window.";
        }
        leaf change-end-time {
          type yang:date-and-time;
          description
            "End date and time of the VPN connectivity service
             maintenance window within the relevant-state window.";
        }
      }
    }

    grouping vpn-node-termination {
      description
        "Node and Network Termination for the VPN Service instance.
         This grouping is used to augment the relevant-state container.";
      list vpn-node-terminations {
        key "hostname vrf-name";
        description
          "List of Node Terminations of interest.";
        leaf hostname {
          type inet:host;
          description
            "The hostname of the network node. This value
              is usually configured on the node by the
              administrator to uniquely identify the node
              in the network.";
        }
        leaf vrf-id {
          type uint32;
          description
            "The VRF id obtained through IPFIX IE234
              ingressVRFID or IE235 egressVRFID.";
        }
        leaf vrf-name {
          type string;
          description
            "The VRF name obtained through IPFIX IE236
              VRFname or BMP peer_up VRF Table Name TLV.";
        }
        leaf route-distinguisher {
          type string;
          description
```

```
            "The BGP route-distinguisher obtained through
              IPFIX IE90 mplsVpnRouteDistinguisher or BMP
              route-monitoring or peer_up message type.";
          }
          leaf-list interface-id {
            type uint32;
            description
              "The interface identifier obtained through
                IPFIX IE10 ingressInterface, IE14
                egressInterface or
                ietf-interfaces:interfaces/interface/if-index.";
          }
          leaf-list interface-name {
            type string;
            description
              "The interface name obtained through
                IPFIX IE82 interfaceName or
                ietf-interfaces:interfaces/interface/name.";
          }
          leaf-list peer-ip {
            type inet:ip-address;
            description
              "The BGP peering IP address learned through
                BMP route-monitoring, peer_up or peer_down
                message type.";
          }
          leaf-list next-hop {
            type inet:ip-address;
            description
              "The BGP next-hop IP address learned through
                BMP route-monitoring message type.";
          }
        }
      }

      augment "/rsn:relevant-state/rsn:service" {
        description
          "Provide extension for the service description,
           specifically for connectivity services to the
           relevant state container.";
        case l2vpn {
          description
            "Layer 2 VPN connectivity service.";
          uses l2vpn-service;
        }
        case l3vpn {
          description
            "Layer 3 VPN connectivity service.";
```

```
        uses l3vpn-service;
      }
    }

    augment "/rsn:relevant-state-notification/rsn:service" {
      description
        "Provide extension for the service description,
         specifically for connectivity services to the
         relevant state notification.";
      case l2vpn {
        description
          "Layer 2 VPN connectivity service.";
        uses l2vpn-service;
      }
      case l3vpn {
        description
          "Layer 3 VPN connectivity service.";
        uses l3vpn-service;
      }
    }

    augment "/rsn:relevant-state/rsn:anomaly" {
      description
        "Provide extension for the service description,
         specifically for connectivity services to the
         relevant state container.";
      uses vpn-node-termination;
    }

    augment "/rsn:relevant-state-notification/rsn:anomaly" {
      description
        "Provide extension for the service description,
         specifically for connectivity services to the
         relevant state notification.";
      uses vpn-node-termination;
    }
  }
  <CODE ENDS>
```

4.4. Apache AVRO Schema

Depending on implementation, a network operator might chose defined YANG models as data models or uses the YANG models as information data models and transform them to another schema format such as [Apache_AVRO] to use as data model for [I-D.ietf-nmop-yang-message-broker-integration] integration.

Shows the entire notification schema of 'ietf-relevant-state.yang' from [I-D.ietf-nmop-network-anomaly-lifecycle], 'ietf-network-anomaly-service-topology.yang' from Section 4.3 and 'ietf-network-anomaly-symptom-cbl.yang' from Section 4.2 as an Apache AVRO schema.

The Apache AVRO schema is decomposed based on YANG groupings as following:

* RelevantStateNotification.avsc is based on 'relevant-state-grouping' defined in 'ietf-relevant-state.yang' with 'ietf.relevant.state.Publisher', 'ietf.relevant.state.Anomaly', 'ietf.relevant.state.VpnNodeTermination' and 'ietf.relevant.state.VpnService' AVRO schema imports.

* Publisher.avsc is based on 'publisher' container defined in 'ietf-relevant-state.yang'.

* Anomaly.avsc is based on 'anomaly-grouping' defined in 'ietf-relevant-state.yang' with 'ietf.relevant.state.Annotator' and 'ietf.relevant.state.Symptom' AVRO schema imports.

* Annotator.avsc is based on 'anotator-grouping' defined in 'ietf-relevant-state.yang'.

* Symptom.avsc is based on 'cbl-symptom' defined in 'ietf-network-anomaly-symptom-cbl.yang'.

* L2VpnService.avsc, L2VpnServiceContainer.avsc, L3VpnService.avsc and L3VpnServiceContainer.avsc is based on 'vpn-service' defined in 'ietf-network-anomaly-service-topology.yang'.

* VpnTermination.avsc is based on 'vpn-termination' defined in 'ietf-network-anomaly-service-topology.yang'.

<CODE BEGINS> file "RelevantStateNotification@2025-11-15.avsc"

```
{
  "type": "record",
  "name": "RelevantStateNotification",
  "namespace": "ietf.relevant.state",
  "fields": [
    {
      "name": "id",
      "type": {
        "type": "string",
        "logicalType": "uuid"
      },
      "doc": "Unique ID of the relevant state. It is unique in the scope of the Label
Store."
     },
```

```
       {
         "name": "uri",
         "type": ["null", "string"],
         "default": null,
         "doc": "URI to visualize the analytical metrics of the relevant-state."
       },
       {
         "name": "description",
         "type": ["null", "string"],
         "default": null,
         "doc": "Textual description of the fault."
       },
       {
         "name": "startTime",
         "type": {
           "type": "long",
           "logicalType": "timestamp-millis"
         },
         "doc": "Date and time indicating the beginning of the problem."
       },
       {
         "name": "endTime",
         "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
         "default": null,
         "doc": "Date and time indicating the end of the problem."
       },
       {
         "name": "strategy",
         "type": ["null", "string"],
         "default": null,
         "doc": "Name of the strategy that detected the relevant state."
       },
       {
         "name": "confidenceScore",
         "type": ["null", "int"],
         "default": null,
         "doc": "Score between 0 and 100 indicating how confident were the detectors in r
elation to the overall relevant state."
       },
       {
         "name": "concernScore",
         "type": "int",
         "doc": "Score between 0 and 100 indicating the degree of concern in relation to
the overall relevant state."
       },
       {
         "name": "anomaly",
         "type": {
           "type": "array",
           "items": "ietf.relevant.state.Anomaly"
```

```
        },
        "doc": "List of anomalies that are part of the relevant state."
      },
      {
        "name": "service",
        "type": [
          "null",
          "ietf.relevant.state.L2VpnServiceContainer",
          "ietf.relevant.state.L3VpnServiceContainer"
        ],
        "default": null,
        "doc": "List of services of interest. The type of the service can be extended in
 the future."
      },
      {
        "name": "publisher",
        "type": "ietf.relevant.state.Publisher",
        "doc": "Name of the system which published the relevant-state notification."
      }
    ]
  }
  <CODE ENDS>

  <CODE BEGINS> file "Anomaly@2025-11-15.avsc"
  {
    "type": "record",
    "name": "Anomaly",
    "namespace": "ietf.relevant.state",
    "fields": [
      {
        "name": "id",
        "type": {
          "type": "string",
          "logicalType": "uuid"
        },
        "doc": "Unique identifier of the anomaly."
      },
      {
        "name": "revision",
        "type": "int",
        "doc": "Revision of the anomaly metadata object."
      },
      {
        "name": "uri",
        "type": ["null", "string"],
        "default": null,
        "doc": "URI to visualize the analytical metrics of the anomaly."
      },
      {
```

```
          "name": "stage",
          "type":  {
              "type": "enum",
              "name": "Stage",
              "symbols": [
                "detection",
                "validation",
                "refinement"
              ]
            },
          "doc": "Stage of the anomaly."
        },
        {
          "name": "description",
          "type": ["null", "string"],
          "default": null,
          "doc": "Textual description of the anomaly."
        },
        {
          "name": "startTime",
          "type": {"type": "long", "logicalType": "timestamp-millis"},
          "doc": "Date and time indicating the beginning of the anomaly."
        },
        {
          "name": "endTime",
          "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
          "default": null,
          "doc": "Date and time indicating the end of the anomaly."
        },
        {
          "name": "confidenceScore",
          "type": ["null", "int"],
          "default": null,
          "doc": "Score between 0 and 100 indicating how confident was the detector while
considering the given anomaly as part of the relevant event."
        },
        {
          "name": "pattern",
          "type": [
            "null",
            {
              "type": "enum",
              "name": "Pattern",
              "symbols": [
                "drop",
                "spike",
                "mean_shift",
                "seasonality_shift",
                "trend",
```

```
              "other"
            ],
            "doc": "Pattern describes the type of pattern that was detected by the annot
ator (e.g. spike, drop, mean_shift, etc.)."
          }
        ],
        "default": null,
        "doc": "Pattern describes the type of pattern that was detected by the annotator
. This field is optional."
      },
      {
        "name": "annotator",
        "type": "ietf.relevant.state.Annotator",
        "doc": "Annotator represents the entity that produced the annotation."
      },
      {
        "name": "symptom",
        "type": ["null", "ietf.relevant.state.Symptom"],
        "default": null,
        "doc": "It specifies the symptom for the anomaly."
      },
      {
        "name": "vpnNodeTerminations",
        "type": {
          "type": "array",
          "items": "ietf.relevant.state.VpnNodeTermination"
        },
        "doc": "List of Node Terminations of interest."
      }
    ]
  }
  <CODE ENDS>
```

```
   <CODE BEGINS> file "Publisher@2025-11-15.avsc"
   {
     "type": "record",
     "name": "Publisher",
     "namespace": "ietf.relevant.state",
     "fields": [
       {
         "name": "id",
         "type": {
           "type": "string",
           "logicalType": "uuid"
         },
         "doc": "Unique ID of the system which published the relevant-state notification.
"
       },
       {
         "name": "name",
         "type": "string",
         "doc": "Name of the system which published the relevant-state notification."
       },
       {
         "name": "version",
         "type": [
           "null",
           {
             "type": "string"
           }
         ],
         "default": null,
         "doc": "Version of the system which published the relevant-state notification.."
       }
     ]
   }
   <CODE ENDS>

   <CODE BEGINS> file "Annotator@2025-11-15.avsc"
   {
     "type": "record",
     "name": "Annotator",
     "namespace": "ietf.relevant.state",
     "fields": [
       {
         "name": "id",
         "type": [
           "null",
           {
             "type": "string",
             "logicalType": "uuid"
           }
```

```
          ],
          "default": null,
          "doc": "Unique ID of the annotator (either user or algorithm)."
        },
        {
          "name": "name",
          "type": "string",
          "doc": "Name of the annotator (either user or algorithm)."
        },
        {
          "name": "annotatorType",
          "type": [
            "null",
            {
              "type": "enum",
              "name": "AnnotatorType",
              "symbols": ["human", "algorithm"],
              "doc": "An annotator can be either a human user or a programmatic entity, su
ch as an algorithm."
            }
          ],
          "default": null,
          "doc": "AnnotatorType specifies the type of the annotator."
        },
        {
          "name": "version",
          "type": [
            "null",
            {
              "type": "string"
            }
          ],
          "default": null,
          "doc": "Version of the annotator."
        }
      ]
    }
    <CODE ENDS>

    <CODE BEGINS> file "Symptom@2025-11-15.avsc"
    {
      "type": "record",
      "name": "Symptom",
      "namespace": "ietf.relevant.state",
      "fields": [
        {
          "name": "id",
          "type": {
            "type": "string",
```

```
            "logicalType": "uuid"
          },
          "doc": "Unique identifier of the symptom type."
        },
        {
          "name": "concernScore",
          "type": "int",
          "doc": "Score between 0 and 100 indicating the degree of concern in relation to
the specific symptom."
        },
        {
          "name": "action",
          "type": ["null", "string"],
          "default": null,
          "doc": "Action associated with the symptom."
        },
        {
          "name": "reason",
          "type": ["null", "string"],
          "default": null,
          "doc": "Reason associated with the symptom."
        },
        {
          "name": "trigger",
          "type": ["null", "string"],
          "default": null,
          "doc": "Trigger associated with the symptom."
        },
        {
          "name": "networkPlane",
          "type": [
            "null",
            {
              "type": "enum",
              "name": "NetworkPlane",
              "symbols": ["management", "control", "forwarding"],
              "doc": "Network Plane affected by the symptom."
            }
          ],
          "default": null,
          "doc": "Network Plane affected by the symptom."
        },
        {
          "name": "template",
          "type": ["null", "string"],
          "default": null,
          "doc": "Name of the template that detected the symptom."
        },
        {
```

```
        "name": "season",
        "type": [
          "null",
          {
            "type": "enum",
            "name": "Season",
            "symbols": ["workday", "holiday"]
          }
        ],
        "default": null,
        "doc": "Associated season. [Note: Other seasons may be added in the future, such
 as weekend.]"
      }
    ]
  }
  <CODE ENDS>

  <CODE BEGINS> file "L2VpnServiceContainer.avsc@2025-11-15.avsc"
  {
    "type": "record",
    "name": "L2VpnServiceContainer",
    "namespace": "ietf.relevant.state",
    "fields": [
      {
        "name": "l2VpnService",
        "type": {
          "type": "array",
          "items": "ietf.relevant.state.L2VpnService"
        },
        "doc": "List of the Layer 2 VPN connectivity services."
      }
    ],
    "doc": "Container for Layer 2 VPN service list."
  }
  <CODE ENDS>

  <CODE BEGINS> file "L2VpnService.avsc@2025-11-15.avsc"
  {
    "type": "record",
    "name": "L3VpnService",
    "namespace": "ietf.relevant.state",
    "fields": [
      {
        "name": "vpnId",
        "type": "string",
        "doc": "Unique ID of the VPN connectivity service."
      },
      {
        "name": "uri",
```

```
        "type": ["null", "string"],
        "default": null,
        "doc": "URI to visualize the VPN service inventory."
      },
      {
        "name": "vpnName",
        "type": ["null", "string"],
        "default": null,
        "doc": "Name of the VPN connectivity service."
      },
      {
        "name": "siteIds",
        "type": ["null", {"type": "array", "items": "string"}],
        "default": null,
        "doc": "List of unique site IDs of the VPN connectivity service."
      },
      {
        "name": "changeId",
        "type": ["null", {"type": "string", "logicalType": "uuid"}],
        "default": null,
        "doc": "Unique identifier of VPN connectivity service maintenance window within
the relevant-state window."
      },
      {
        "name": "changeStartTime",
        "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
        "default": null,
        "doc": "Start date and time of the VPN connectivity service window within the re
levant-state window."
      },
      {
        "name": "changeEndTime",
        "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
        "default": null,
        "doc": "End date and time of the VPN connectivity service window within the rele
vant-state window."
      }
    ]
  }
  <CODE ENDS>
```

```
<CODE BEGINS> file "L3VpnServiceContainer.avsc@2025-11-15.avsc"
{
  "type": "record",
  "name": "L3VpnServiceContainer",
  "namespace": "ietf.relevant.state",
  "fields": [
    {
      "name": "l3VpnService",
      "type": {
        "type": "array",
        "items": "ietf.relevant.state.L3VpnService"
      },
      "doc": "List of the Layer 3 VPN connectivity services."
    }
  ],
  "doc": "Container for Layer 3 VPN service list."
}
<CODE ENDS>

<CODE BEGINS> file "L3VpnService.avsc@2025-11-15.avsc"
{
  "type": "record",
  "name": "L3VpnService",
  "namespace": "ietf.relevant.state",
  "fields": [
    {
      "name": "vpnId",
      "type": "string",
      "doc": "Unique ID of the VPN connectivity service."
    },
    {
      "name": "uri",
      "type": ["null", "string"],
      "default": null,
      "doc": "URI to visualize the VPN service inventory."
    },
    {
      "name": "vpnName",
      "type": ["null", "string"],
      "default": null,
      "doc": "Name of the VPN connectivity service."
    },
    {
      "name": "siteIds",
      "type": ["null", {"type": "array", "items": "string"}],
      "default": null,
      "doc": "List of unique site IDs of the VPN connectivity service."
    },
```

```
       {
         "name": "changeId",
         "type": ["null", {"type": "string", "logicalType": "uuid"}],
         "default": null,
         "doc": "Unique identifier of VPN connectivity service maintenance window within
the relevant-state window."
       },
       {
         "name": "changeStartTime",
         "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
         "default": null,
         "doc": "Start date and time of the VPN connectivity service window within the re
levant-state window."
       },
       {
         "name": "changeEndTime",
         "type": ["null", {"type": "long", "logicalType": "timestamp-millis"}],
         "default": null,
         "doc": "End date and time of the VPN connectivity service window within the rele
vant-state window."
       }
     ]
   }
   <CODE ENDS>

   <CODE BEGINS> file "VpnNodeTermination@2025-11-15.avsc"
   {
     "type": "record",
     "name": "VpnNodeTermination",
     "namespace": "ietf.relevant.state",
     "fields": [
       {
         "name": "hostname",
         "type": ["null", "string"],
         "default": null,
         "doc": "The hostname of the network node."
       },
       {
         "name": "routeDistinguisher",
         "type": ["null", "string"],
         "default": null,
         "doc": "The BGP route-distinguisher obtained through IPFIX or BMP."
       },
       {
         "name": "peerIp",
         "type": {"type": "array", "items": "string"},
         "doc": "The BGP peering IP address."
       },
       {
         "name": "nextHop",
         "type": {"type": "array", "items": "string"},
```

```
        "doc": "The BGP next-hop IP address."
      },
      {
        "name": "vrfId",
        "type": ["null", "long"],
        "default": null,
        "doc": "The VRF identifier."
      },
      {
        "name": "vrfName",
        "type": ["null", "string"],
        "default": null,
        "doc": "The VRF name."
      },
      {
        "name": "interfaceId",
        "type": {"type": "array", "items": "long"},
        "doc": "The interface identifier."
      },
      {
        "name": "interfaceName",
        "type": {"type": "array", "items": "string"},
        "doc": "The interface name."
      }
    ]
  }
  <CODE ENDS>
```

5. IANA Considerations

This document registers the following two namespace URIs in the IETF XML Registry [RFC3688]:

URI: urn:ietf:params:xml:ns:yang:ietf-network-anomaly-symptom-cbl

Registrant Contact: The IESG.

XML: N/A; the requested URI is an XML namespace.

URI: urn:ietf:params:xml:ns:yang:ietf-network-anomaly-service-topology

Registrant Contact: The IESG.

XML: N/A; the requested URI is an XML namespace.

This document registers the following two YANG modules in the YANG Module Names registry [RFC3688]:

Name: ietf-network-anomaly-symptom-cbl

Namespace: urn:ietf:params:xml:ns:yang:ietf-network-anomaly-symptom-cbl

Prefix: smcblsymptom

Reference: RFC XXXX

Name: ietf-network-anomaly-service-topology

Namespace: urn:ietf:params:xml:ns:yang:ietf-network-anomaly-service-topology

Prefix: smtopology

Reference: RFC XXXX

6. Security Considerations

This section is modeled after the template described in Section 3.7 of [I-D.ietf-netmod-rfc8407bis].

The "ietf-network-anomaly-symptom-cbl" and "ietf-network-anomaly-service-topology" YANG modules defines two data models that are designed to be accessed via YANG-based management protocols, such as NETCONF [RFC6141] and RESTCONF [RFC8040]. These protocols have to use a secure transport layer (e.g., SSH [RFC4252], TLS [RFC8446], and QUIC [RFC9000]) and have to use mutual authentication.

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., "config true", which is the default). All writable data nodes are likely to be reasonably sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) and delete operations to these data nodes without proper protection or authentication can have a negative effect on network operations. The following subtrees and data nodes have particular sensitivities/vulnerabilities:

"There are no particularly sensitive writable data nodes."

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. Specifically, the following subtrees and data nodes have particular sensitivities/ vulnerabilities:

"There are no particularly sensitive readable data nodes."

7. Implementation status

This section provides pointers to existing open source implementations of this draft. Note to the RFC-editor: Please remove this before publishing.

7.1. Antagonist

A tool called Antagonist has been implemented and refined during the IETF 119 and 120 hackathons, in order to validate the application of the YANG models defined in this draft. Antagonist provides visual support for two important use cases in the scope of this document:

* the generation of a ground truth in relation to Symptoms and Problems in timeseries data

* the visual validation of results produced by automated network anomaly detection tools.

The open source code can be found here: [Antagonist]

7.2. Cosmos Bright Lights

A real-time streaming based Service Disruption Detection System has been deployed in Swisscom production as a proof of concept in June 2024 monitoring approximate >13’000 L3 VPN’s concurrently. The Apache AVRO schema described in Section 4.4 is being implemented in April 2025 in the development enviroment and considered to be deployed in June 2025 in production.

8. Acknowledgements

The authors would like to thank , for his review and valuable comment.

The authors would like to thank Antonio Roberto for his contribution to the ideas in this draft and Reshad Rahman, Mohamed Boucadair and Ruediger Geib for his review and valuable comments.

9. References

9.1. Normative References

[Antagonist]
Riccobene, V., Du, W., Graf, T., and H. Huang Feng, "Antagonist: Anomaly tagging on historical data", <https://github.com/vriccobene/antagonist>.

[I-D.ietf-nmop-network-anomaly-architecture]
Graf, T., Du, W., Francois, P., and A. H. Feng, "A Framework for a Network Anomaly Detection Architecture", Work in Progress, Internet-Draft, draft-ietf-nmop-network-anomaly-architecture-06, 21 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-network-anomaly-architecture-06>.

[I-D.ietf-nmop-network-anomaly-lifecycle]
Riccobene, V., Graf, T., Du, W., and A. H. Feng, "An Experiment: Network Anomaly Detection Lifecycle", Work in Progress, Internet-Draft, draft-ietf-nmop-network-anomaly-lifecycle-04, 21 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-network-anomaly-lifecycle-04>.

[I-D.ietf-nmop-terminology]
Davis, N., Farrel, A., Graf, T., Wu, Q., and C. Yu, "Some Key Terms for Network Fault and Problem Management", Work in Progress, Internet-Draft, draft-ietf-nmop-terminology-23, 18 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-terminology-23>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC4252] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, DOI 10.17487/RFC4252, January 2006, <https://www.rfc-editor.org/info/rfc4252>.

[RFC6141] Camarillo, G., Ed., Holmberg, C., and Y. Gao, "Re-INVITE and Target-Refresh Request Handling in the Session Initiation Protocol (SIP)", RFC 6141, DOI 10.17487/RFC6141, March 2011, <https://www.rfc-editor.org/info/rfc6141>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8969] Wu, Q., Ed., Boucadair, M., Ed., Lopez, D., Xie, C., and L. Geng, "A Framework for Automating Service and Network Management with YANG", RFC 8969, DOI 10.17487/RFC8969, January 2021, <https://www.rfc-editor.org/info/rfc8969>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

9.2. Informative References

[Apache_AVRO] "Apache AVRO", Apache Software Foundation, <https://avro.apache.org/>.

[I-D.ietf-netmod-rfc8407bis] Bierman, A., Boucadair, M., and Q. Wu, "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", Work in Progress, Internet-Draft, draft-ietf-netmod-rfc8407bis-28, 5 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-netmod-rfc8407bis-28>.

[I-D.ietf-nmop-yang-message-broker-integration]
Graf, T. and A. Elhassany, "An Architecture for YANG-Push to Message Broker Integration", Work in Progress, Internet-Draft, draft-ietf-nmop-yang-message-broker-integration-10, 18 January 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-nmop-yang-message-broker-integration-10>.

[I-D.ietf-opsawg-collected-data-manifest]
Claise, B., Quilbeuf, J., Lopez, D., Martinez-Casanueva, I. D., and T. Graf, "A Data Manifest for Contextualized Telemetry Data", Work in Progress, Internet-Draft, draft-ietf-opsawg-collected-data-manifest-10, 20 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-opsawg-collected-data-manifest-10>.

Authors' Addresses

Thomas Graf
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: thomas.graf@swisscom.com

Wanting Du
Swisscom
Binzring 17
CH-8045 Zurich
Switzerland
Email: wanting.du@swisscom.com

Alex Huang Feng
INSA-Lyon
Lyon
France
Email: alex.huang-feng@insa-lyon.fr

Vincenzo Riccobene
Huawei
Dublin
Ireland
Email: vincenzo.riccobene@huawei-partners.com