Common Control and Measurement Plane
Internet-Draft
Intended status: Standards Track
Expires: 11 July 2026

C. Yu
Huawei Technologies
F. Peruzzini
FiberCop
Y. Zheng
China Unicom
I. Busi
Huawei Technologies
A. Guo
Futurewei Technologies
V. Lopez
Nokia
X. Zhao
CAICT
M. Jin
Huawei Technologies
7 January 2026

A YANG Data Model for Resource Performance Monitoring
draft-yu-ccamp-resource-pm-yang-05

Abstract

This document defines a YANG data model for resource Performance Monitoring, applicable to network controllers, which provides the functionalities of retrieval of performance monitoring capabilities, TCA (Threshold Crossing Alert) configuration, current or history performance data retrieval, and performance monitoring task management.

About This Document

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

The latest revision of this draft can be found at https://YuChaode.github.io/draft-yu-ccamp-resource-pm-yang/draft-yu-ccamp-resource-pm-yang.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-yu-ccamp-resource-pm-yang/.

Discussion of this document takes place on the Common Control and Measurement Plane Working Group mailing list (mailto:ccamp@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ccamp/. Subscribe at https://www.ietf.org/mailman/listinfo/ccamp/.

Source for this draft and an issue tracker can be found at https://github.com/YuChaode/draft-yu-ccamp-resource-pm-yang.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Resource performance monitoring is a basic function of network management. By viewing and analyzing the performance data of different resources (such as network element, interface, board, termination point, tunnel termination point) operators can detect the running state of the network in time, quickly resolve real-time problems or identify major risks in advanced , avoiding users' complaints.

According to the business requirements stated in [TMF-518], resource performance monitoring requirements include:

* Retrieval of current and historical performance measurements for network resources

* Distribution of Threshold Crossing Alerts (TCAs) to the collectors of PM data

* Control of performance monitoring in the network: enable/disable of PM collection and TCA generation

Currently, there are some existing documents related to performance monitoring in IETF, but there is no overlap with our current work. The relative monitored objects are summarized in Table 1.

YANG Module	Performance-Monitored Object	Reference
ietf-te-telemetry, ietf-vn-telemetry	TE tunnel, virtual network	[I-D.ietf-teas-actn-pm-telemetry-autonomics]
ietf-network-vpn-pm	VPN service topology	[RFC9375]
ietf-service-pm	Transport network service	[I-D.zheng-ccamp-client-pm-yang]
ietf-lmap-control, ietf-lmap-report	massive measurement agents in broadband service	[RFC8194]

Table 1

[I-D.ietf-teas-actn-pm-telemetry-autonomics] provides a YANG data model that describes performance monitoring and scaling intent mechanisms for TE-Tunnels and Virtual Networks(VNs). [I-D.ietf-opsawg-yang-vpn-service-pm] defines a YANG data model for performance monitoring of both network topology layer and overlay VPN service topology layer. [I-D.zheng-ccamp-client-pm-yang] provides a performance monitoring YANG data model on client signal level. [RFC8194] defines a data model for Large-Scale Measurement Platforms(LMAP), focusing on task capability and configuration of massive measurement agents.

This document defines a YANG module for resource performance monitoring, which defines the capabilities of resource performance monitoring, the tca configuration model of a specific resource. In addition, the sub-module of monitoring task and a few RPCs are defined to support the operations of performance monitoring, such as data retrieval and controlling the monitoring tasks.

The YANG data model defined in this document conforms to the Network Management Datastore Architecture (NMDA) defined in [RFC8342].

1.1. Terminology and Notations

Refer to [RFC7446] and [RFC7581] for the key terms used in this document. The following terms are defined in [RFC7950] and are not redefined here: * client * server * augment * data model * data node

The following terms are defined in [RFC6241] and are not redefined here: * configuration data * state data

The following terms are defined in [RFC8454] and are not redefined here: * CMI * MPI * MDSC * CNC * PNC

To Be Added: some explanation of performance indicator

1.2. Tree Diagram

A simplified graphical representation of the data model is used in Section 3 of this document. The meaning of the symbols in these diagrams are defined in [RFC8340].

1.3. Prefix in Data Node Names

In this document, names of data nodes and other data model objects are prefixed using the standard prefix associated with the corresponding YANG imported modules, as shown in the following table.

Prefix	Yang Module	Reference
rpm	ietf-resource-pm	RFC XXXX
rpm-types	ietf-resource-pm-types	RFC XXXX
yang	ietf-yang-types	[RFC6991]

Table 2: Prefixes and corresponding YANG modules

RFC Editor Note: Please replace XXXX with the RFC number assigned to this document.

2. YANG Data Model for Resource Performance Monitoring

2.1. Capabilities of Resource Performance Monitoring

Supported Resources and Corresponding PM Capabilities A generic "resources" structure is defined to describe the objects that could be monitored:

* "resource": the identifier of a monitored object

* "resource-type": indicate what exact kind of resource is

* "holding time": the longest time period that performance data could be monitored

* "indicator name": the indicators that could be monitored for the resource

* "sub-resources": the identifier of the performace monitoring point of this resource (e.g., if the resource is a network element, the sub resources may be a list of termination points)

```
module: ietf-resource-pm
   +--rw performance-monitoring
      +--rw resources
         +--rw resource-list* [resource]
            +--rw resource             union
            +--ro resource-type?       identityref
            +--ro holding-time?     uint8
            +--ro indicator-name*   string
            +--ro sub-resources*    -> ../resource
```

2.1.1. Introduction of Performance Indicators

It is impossible to list all the PM indicator exhaustively, even if ITU-T and IETF has done a lot of work. Some new performance indicators would be raised once there are some new requirements and technologies. So in this document we would like to provide String type rather than an explicit type for performance indicator, to have a better compatibility for future extension. Then if there are some new indicators, there is no need to revise this document or create a branch of documents to standardize the PM indicators.

For optical network, a performance management information table in the session10.2 of [ITU-T_G.874] lists all the PM current data and history data collected on the EMF(Equipment Management Function). A few indicators of different resources are listed below as an example.

// To be added: performance indicators of different resources in optical network.

For IP network, a few indicators of different resources are listed below as an example. For a network element, following are possible indicators: * CPU usage. * Memory usage. * Total active routes. * Total active mac entries.

For an interface, following are possible indicators: * Ingress unicast packets. * Ingress multicast packets. * Ingress discard packets. * Ingress error packets. * Ingress unknown protocol packets. * Egress unicast packets. * Egress multicast packets. * Egress discard packets. * Egress error packets.

For a board, following are possible indicators: * CPU usage. * Memory usage.

2.2. Threshold Crossing Alert Control

Threshold crossing alert control parameters could be set directly for a resource, or set through applying an existing profile to the resource. Therefore, there are four main requirements for Threshold Crossing Alert Control:

* Creation/retrieval/deletion/updating of TCA profile;

* Enabling/disabling TCA reporting on the resource;

* Configuring TCA on the resource by associating an existing profile;

* Configuring TCA on the resource by detailed parameters.

To satisfy the above requirements, the module defines "tca-management", including the "profile" structure to enable the preset of tca parameters, "tca" structure to describe the tca parameters (directly set or preset by applying profiles) and tca status of a specific resource.

And for the TCA parameters, no matter it is configured directly on the resource or by a preset profile, there should not be any differences. The TCA parameters (tca-indicator) should include:

* Threshold-type: This threshold type is used to indicate when the alert will be triggered. By exceeding the upper bound value, or by below the lower bound value.

* Period: This period is used to indicate the frequency of the data collection.

* Severity: This severity is used to indicate what level of alert would be triggered if cross the threshold.

* Indicator-value: The value of threshold.

* Indicator-value-unit: The unit of threshold value.

In addition, the function of enabling/disabling TCA on the resource can be controlled by the "admin-status" attribute in "tca" node. The pre-defined profiles with unique profile id could be applied to "tca" node, shown as "applied-profiles" in "tca" node.

```
module: ietf-resource-pm
  +--rw performance-monitoring
     +--rw tca-management
     |  +--rw profiles
     |  |  +--rw profile* [profile-id]
     |  |     +--rw profile-id      yang:uuid
     |  |     +--rw profile-name?   string
     |  |     +--rw tca-cfg
     |  |        +--rw tca-indicator* [indicator-name threshold-type period severity]
     |  |           +--rw indicator-name          string
     |  |           +--rw indicator-value         string
     |  |           +--rw indicator-value-unit    string
     |  |           +--rw threshold-type          enumeration
     |  |           +--rw period                  identityref
     |  |           +--rw severity                identityref
     |  +--rw tcas
     |     +--rw tca* [resource]
     |        +--rw resource            -> /performance-monitoring/resources/resource-lis
t/resource
     |        +--ro resource-type?      identityref
     |        +--rw admin-status?       enumeration
     |        +--rw applied-profiles
     |        |  +--rw profile* [profile-id]
     |        |     +--rw profile-id    -> ../../../../../profiles/profile/profile-id
     |        +--rw tca-cfg
     |           +--rw tca-indicator* [indicator-name threshold-type period severity]
     |              +--rw indicator-name          string
     |              +--rw indicator-value         string
     |              +--rw indicator-value-unit    string
     |              +--rw threshold-type          enumeration
     |              +--rw period                  identityref
     |              +--rw severity                identityref
```

2.3. Data Retrieval of Resource Performance Monitoring

2.3.1. Get Current/History Performance Monitoring Data

For the retrieval of current/history performance data, we consider these data are not suitable to define in a data model. Because performance data can be changed frequently and if we follow that approach, according to the requirement of on-change notification in YANG-push [RFC8641], once the performance data changes, the controller should trigger a notification to the northbound system, there would be great number of notifications reported in the big

network.

These two retrieval interfaces are usually invoked on-demand. It is also hard to support retrieving performance data of multiple resources by data model in one request. And for history performance data retrieval, there could be a requirement to specify the start time and end time. It is not quite flexible to support this requirement by data model neither. So we suggest to define an RPC to accomplish these two functions.

```
  rpcs:
    +---x get-pm-data
    |  +---w input
    |  |  +---w resources*                    -> /performance-monitoring/resources/resou
rce-list/resource
    |  |  +---w is-requesting-history-data?   boolean
    |  |  +---w start-time?                    yang:date-and-time
    |  |  +---w end-time?                      yang:date-and-time
    |  +--ro output
    |     +--ro pm-data
    |        +--ro pm-data-list* [resource]
    |           +--ro resource          -> /performance-monitoring/resources/resource-lis
t/resource
    |           +--ro task-id*          yang:uuid
    |           +--ro collect-time?     yang:date-and-time
    |           +--ro resource-type?    identityref
    |           +--ro indicator-data
    |              +--ro indicator-data-list* [indicator-name]
    |                 +--ro indicator-name          string
    |                 +--ro indicator-value?        string
    |                 +--ro indicator-value-unit?   string
```

2.3.2. Get Profile Associated Resources

For the TCA related definition, it can be found in the previous Section (Section2.2). A TCA profile can be associated with a lot of resources, so we don’t defined a resource list under the profile instance to avoid reporting some unnecessary notifications while the resource instances in this list have been changed. We define an RPC operation to support this flexible retrieval request.

```
  rpcs:
    +---x get-profile-associated-resources
       +---w input
       |  +---w profile-id?   -> /performance-monitoring/tca-management/profiles/profile/
profile-id
       +--ro output
          +--ro resource-list*   -> /performance-monitoring/resources/resource-list/resou
rce
```

2.4. Controlling of Resource Performance Monitoring

2.4.1. Clear Performance Monitoring Data of Specific Resources

We define an RPC to enable clearing the performance monitoring data of specified resources. If there are any resources failed to clear performance monitoring data, their identifier should be returned by the failed-resources leaf list. An empty list indicates that this operation was performed successfully.

```
  rpcs:
    +---x clear-performance-monitoring-data
    |  +---w input
    |  |  +---w resources*   -> /performance-monitoring/resources/resource-list/resource
    |  +--ro output
    |     +--ro failed-resources*   -> /performance-monitoring/resources/resource-list/re
source
```

2.4.2. Enable/Disable Performance Monitoring

To enable/disable performance monitoring data, we introduce a monitor task to do this control. In the monitoring task, the resource, the monitored period, the monitored indicators could be set. The disabling, enabling operation can be satisfied by changing the admin status which includes disabled, enabled. The change's result will affect the task status accordingly.

```
      module: ietf-resource-pm
         +--rw performance-monitoring
            +--rw monitor-tasks
            |  +--rw monitor-task* [task-id]
            |     +--rw task-id          yang:uuid
            |     +--rw resource-id?     leafref
            |     +--ro resource-type?   identityref
            |     +--rw task-name?       string
            |     +--rw admin-status?    enumeration
            |     +--ro task-status?     enumeration
            |     +--rw task-cfg
            |        +--rw period?       identityref
            |        +--rw indicators
            |           +--rw indicator* [indicator-name]
            |              +--rw indicator-name          string
            |              +--rw indicator-value-unit?   string
```

3. Resource Performance Monitoring Tree Diagram

```
   =============== NOTE: '\' line wrapping per RFC 8792 ================

   module: ietf-resource-pm
     +--rw performance-monitoring
        +--rw resources
        |  +--rw resource-list* [resource]
        |     +--rw resource          union
        |     +--ro resource-type?    identityref
        |     +--ro holding-time?     uint8
        |     +--ro indicator-name*   string
        |     +--ro sub-resources*    -> ../resource
        +--rw tca-management
        |  +--rw profiles
        |  |  +--rw profile* [profile-id]
        |  |     +--rw profile-id      yang:uuid
        |  |     +--rw profile-name?   string
        |  |     +--rw tca-cfg
        |  |        +--rw tca-indicator*
        |  |                [indicator-name threshold-type period \
                                                            severity]
        |  |           +--rw indicator-name          string
        |  |           +--rw indicator-value         string
        |  |           +--rw indicator-value-unit    string
        |  |           +--rw threshold-type          enumeration
        |  |           +--rw period                  identityref
        |  |           +--rw severity                identityref
        |  +--rw tcas
        |     +--rw tca* [resource]
        |        +--rw resource            leafref
        |        +--ro resource-type?      identityref
        |        +--rw admin-status?       enumeration
        |        +--rw applied-profiles
        |        |  +--rw profile* [profile-id]
        |        |     +--rw profile-id    leafref
        |        +--rw tca-cfg
        |           +--rw tca-indicator*
        |                   [indicator-name threshold-type period \
                                                            severity]
        |              +--rw indicator-name          string
        |              +--rw indicator-value         string
        |              +--rw indicator-value-unit    string
        |              +--rw threshold-type          enumeration
        |              +--rw period                  identityref
        |              +--rw severity                identityref
        +--rw monitor-tasks
           +--rw monitor-task* [task-id]
              +--rw task-id          yang:uuid
              +--rw resource?        leafref
```

```
            +--ro resource-type?   identityref
            +--rw task-name?       string
            +--rw admin-status?    enumeration
            +--ro task-status?     enumeration
            +--rw task-cfg
               +--rw period?       identityref
               +--rw indicators
                  +--rw indicator* [indicator-name]
                     +--rw indicator-name          string
                     +--rw indicator-value-unit?   string

     rpcs:
       +---x get-pm-data
       |  +---w input
       |  |  +---w resources*                     leafref
       |  |  +---w is-requesting-history-data?   boolean
       |  |  +---w start-time?                    yang:date-and-time
       |  |  +---w end-time?                      yang:date-and-time
       |  +--ro output
       |     +--ro pm-data
       |        +--ro pm-data-list* [resource]
       |           +--ro resource          leafref
       |           +--ro task-id*          yang:uuid
       |           +--ro collect-time?     yang:date-and-time
       |           +--ro resource-type?    identityref
       |           +--ro indicator-data
       |              +--ro indicator-data-list* [indicator-name]
       |                 +--ro indicator-name          string
       |                 +--ro indicator-value?        string
       |                 +--ro indicator-value-unit?   string
       +---x clear-performance-monitoring-data
       |  +---w input
       |  |  +---w resources*   leafref
       |  +--ro output
       |     +--ro failed-resources*   leafref
       +---x get-profile-associated-resources
          +---w input
          |  +---w profile-id?   leafref
          +--ro output
             +--ro resource-list*   leafref
```

Figure 1: Resource Performance Monitoring tree diagram

4. YANG Model for Resource Performance Monitoring

```
<CODE BEGINS> file "ietf-resource-pm@2025-07-04.yang"
module ietf-resource-pm {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-resource-pm";
  prefix rpm;

  import ietf-yang-types {
    prefix "yang";
  }

  import ietf-resource-pm-types {
    prefix "rpm-types";
  }

  organization
    "IETF CCAMP Working Group";

  contact
    "WG Web:   <https://datatracker.ietf.org/wg/ccamp/>
     WG List:  <mailto:ccamp@ietf.org>

     Editor:   Chaode Yu
               <yuchaode@huawei.com>

     Editor:   Fabio Peruzzini
               <fabio.peruzzini@fibercop.com>

     Editor:   Yanlei Zheng
               <zhengyanlei@chinaunicom.cn>

     Editor:   Victor Lopez
               <victor.lopez@nokia.com>

     Editor:   Italo Busi
               <italo.busi@huawei.com>

     Editor:   Aihua Guo
               <aihuaguo.ietf@gmail.com>

     Editor:   Xing Zhao
               <zhaoxing@caict.ac.cn>

     Editor:   Mingshuang Jin
               <jinmingshuang@huawei.com>";

  description
    "This module defines a model for resource performance
    monitoring.
```

```
      The model fully conforms to the Network Management
      Datastore Architecture (NMDA).

      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).

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

      The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL
      NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'NOT RECOMMENDED',
      'MAY', and 'OPTIONAL' in this document are to be interpreted as
      described in BCP 14 (RFC 2119) (RFC 8174) when, and only when,
      they appear in all capitals, as shown here.";

    revision 2025-07-04 {
      description  "Initial revision.";
    }

    container performance-monitoring {
      description
        "the root node.";

      uses resource-info-grouping;
      uses tca-management-grouping;
      uses monitoring-tasks-grouping;
    }

    grouping resource-info-grouping {
      description
        "grouping of resources' PM capabilities related information";

      container resources {
        description
          "resources' PM capabilities related information";

        list resource-list {
          description
            "list of resource instances";
          key resource;
```

```
            leaf resource {
              type union {
                type instance-identifier {
                  require-instance false;
                }
                type yang:object-identifier;
                type string;
                type yang:uuid;
              }
            }

            leaf resource-type {
              type identityref {
                base rpm-types:resource-type;
            }
            config false;
              description
              "the type of resource, such as NE, board or port";
          }

           leaf holding-time {
              description
                "Contains the time period in hours within which 24h PM
                 data records and 15min PM data records may be retrieved.
                 If the domain controller does not store PM data it is the
                 time supported in the NE";
            config false;
            type uint8;
            units "hour";
           }

          leaf-list indicator-name {
             type string;
             config false;
          }

          leaf-list sub-resources {
            description
               "the identifier of the performace monitoring point of this
               resource. If the resource is a NE, the sub-resource should
               be termination point. If the resource is a termination
               point contained in this NE, the sub-resource should be the
               logic channel on this TP";
           config false;
           type leafref {
             path "../rpm:resource";
           }
         }
```

```
    }
  }

  }

    grouping monitoring-tasks-grouping {
      description
        "grouping of performance monitoring task";

      container monitor-tasks {
        description
          "Information of PM tasks";

        list monitor-task {
          description
            "monitoring task list";

          key task-id;
          uses task-instance-grouping;
        }
      }
    }

    grouping task-instance-grouping {
      description
        "grouping of performance monitoring task. In this monitoring
        task, the client can specify a resource to run a monitor task
        and what kind of performance data need to be monitored.";

      leaf task-id {
        description
          "identifier of the performance task";
        type yang:uuid;
      }

      leaf resource {
        description
          "the identifier of network resource on which the performance
          monitoring task is running";

        type leafref {
          path "/rpm:performance-monitoring/rpm:resources" +
          "/rpm:resource-list/rpm:resource";
        }
      }

      leaf resource-type {
        description
```

```
            "the type of resource, such as NE, board or port";

          config false;
          type identityref {
            base rpm-types:resource-type;
          }
        }

        leaf task-name {
          description
            "the name of monitoring task";

          type string;
        }

        leaf admin-status {
          description
            "it is used to control enbling/disabling PM task";

          type enumeration {
            enum enabled {
              description
                "it is used to enable the pm task, if the task is enabled,
                the task-staus should be running";
            }

            enum disabled {
              description
                "it is used to disenable the pm task, if the task is
                disabled, the task-staus should be suspended";
            }
          }
        }

        leaf task-status {
          config false;
          description
            "the status of monitoring task.";

          type enumeration {
            enum running;

            enum suspended;

            enum abnormal;
          }
        }
```

```
      uses task-configuration-grouping;
    }

    grouping task-configuration-grouping {
      description
        "grouping of pm task configuration";

      container task-cfg {
        description
          "Configuration of the monitoring task";

        leaf period {
          description
            "it is used to indicate the interval as per monitoring
            task";
          type identityref {
            base rpm-types:period;
          }
        }

        container indicators {
          description
            "performance indicators";

          list indicator {
            description
              "list of PM indicators to be monitored";

            key indicator-name;
            uses indicator-grouping;
          }
        }
      }
    }

    grouping indicator-grouping {
      description
        "grouping of a monitoring indicator instance";

      leaf indicator-name {
        description
          "performance indicator's name";

        type string;
      }

      leaf indicator-value-unit {
        description
```

```
          "unit of indicator value";

        type string;
      }
    }

    grouping tca-management-grouping {
      description
        "grouping of configuration and management for Threshol Crossing
        Alert";

      container tca-management {
        description
          "configuration and management for Threshol Crossing Alert";

        container profiles {
          description
            "the TCA profile in the whole network";

          list profile {
            description
              "List of TCA profile instances";

            key "profile-id";
            uses tca-profile-grouping;
          }
        }

        container tcas {
          description
            "TCA configuration on the network resources.";

          list tca {
            description
              "List of TCA configuration instances";

            key "resource";

            leaf resource {
              description
                "the identifier of network resource on which threshold
                is configured for TCA purpose";

              type leafref {
                path "/rpm:performance-monitoring/rpm:resources" +
                "/rpm:resource-list/rpm:resource";
              }
            }
```

```
            leaf resource-type {
              description
                "the type of resource, such as NE, board or termination
                point";

              config false;

              type identityref {
                base rpm-types:resource-type;
              }
            }

            leaf admin-status {
              description
                "it is used to control the validity of threshold";
              type enumeration {
                enum enabled {
                description
                  "if the admin-status of TCA configuration is enabled,
                  the threshold is effective";
                }

                enum disabled {
                  description
                    "if the admin-status of TCA configuration is
                    disabled, the threshold is not effective";
                }
              }
            }

            uses tca-grouping;
          }
        }
      }
    }

    grouping tca-profile-grouping {
      description
        "grouping of TCA profile instance";

      leaf profile-id {
        description
          "identifier of threshold crossing alerrt profile";

        type yang:uuid;
      }

      leaf profile-name {
```

```
        description
          "Name of the threshold crossing alerrt profile";

        type string;
      }

      container tca-cfg {
        description
          "detailed TCA configuration in a profile";

        list tca-indicator {
          description
            "list of TCA configuration.";

          key "indicator-name threshold-type period severity";
          uses tca-indicator-grouping;
        }
      }
    }

    grouping tca-indicator-grouping {
      description
        "grouping for detail TCA configuration";
      leaf indicator-name {
        type string;
        description
          "name of the indicator";
      }

      leaf indicator-value {
        mandatory true;
        type string;
        description
          "treshold value of the indicator";
      }

      leaf indicator-value-unit {
        mandatory true;
        type string;
        description
          "unit of indicator's value";
      }

      leaf threshold-type {
        description
          "it is used to indicate the trigger/clearing condition of
          alert upon the threshold";
```

```
          type enumeration {
            enum upperbound-trigger {
              description
                "If the performace data exceeds this threshold value,
                a TCA will be triggered";
            }
            enum lowerbound-trigger {
              description
                "if the performace data is lower than this threshold
                value, a TCA will be triggered";
            }
            enum upperbound-clear {
              description
                "If the performance data is not longer bigger than this
                threshold value, the TCA triggered before will be cleared
                by system automatically";
            }
            enum lowerbound-clear {
              description
                "If the performance data is not longer lower than this
                threshold value, the TCA triggered before will be cleared
                by system automatically";
            }
          }
        }

        leaf period {
          description
            "it is used to indicate the interval as per monitoring task";

          type identityref {
            base rpm-types:period;
          }
        }

        leaf severity {
          description
            "it is used to indicate what severity level of alert would be
            triggered if not confirms to the threashold";

          type identityref {
            base rpm-types:severity;
          }
        }
      }

      grouping tca-grouping {
        description
```

```
        "grouping of TCA configuration";

      container applied-profiles {
          description
            "Information of applied TCA profiles on this resource";

        list profile {
            description
              "list of applied TCA profile";

          key "profile-id";

          leaf profile-id {
            description
              "identifier of the applied TCA profile";

            type leafref {
              path "../../../../../rpm:profiles/rpm:profile" +
              "/rpm:profile-id";
            }
          }
        }
      }
      container tca-cfg {
        description
          "detailed configuration of TCA";
        list tca-indicator {
           description
             "list of tca indicator configuration";
        key "indicator-name threshold-type period severity";
        uses tca-indicator-grouping;
        }
      }
    }

    rpc get-pm-data {
      input {

        leaf-list resources {
          description
            "the identifier of resources from which performace
            data is collected";

          type leafref {
            path "/rpm:performance-monitoring/rpm:resources" +
            "/rpm:resource-list/rpm:resource";
          }
        }
```

```
         leaf is-requesting-history-data {
           description
           "true indicate this is a request for historic data, then start-time and end-ti
me should be assigned";
           type boolean;
         }
         leaf start-time {
           description
           "the starttime of performance data needed to be retrieved";

           type yang:date-and-time;
         }

         leaf end-time {
           description
           "the endtime of performance data needed to be retrieved";

           type yang:date-and-time;
         }
       }
       output {
         container pm-data {
           description
             "result of historic performance data";

           list pm-data-list {
             description
               "list of historic performance data";

             key resource;
             uses pm-data-instance-grouping;
           }
         }
       }
     }

     rpc clear-performance-monitoring-data {
           description
           "This operation clears (reset) the PM registers for a list of
           Measurement Points. Within the request for each Measurement
           Point, it is possible to specify the granularity (15min, 24hr,
           NA) and location (nearEnd and/or farEnd and/or bidirectional)
           for the PM registers that are to be reset.";
       input {

         leaf-list resources {
           description
             "the identifier of measurement points to clear PM data";
```

```
            type leafref {
              path "/rpm:performance-monitoring/rpm:resources" +
              "/rpm:resource-list/rpm:resource";
            }
          }
        }
        output {
          leaf-list failed-resources {
            description
              "the identifier of measurement points which are failed to
              clear PM data. An empty list indicates that the total
              request was successful.";

            type leafref {
              path "/rpm:performance-monitoring/rpm:resources" +
              "/rpm:resource-list/rpm:resource";
            }
          }
        }
      }

      rpc get-profile-associated-resources {
        input {

          leaf profile-id {
            description
              "the identifier of profile which the client want to
              retrieve";

            type leafref {
              path "/rpm:performance-monitoring/rpm:tca-management"
              + "/rpm:profiles/rpm:profile/rpm:profile-id";
            }
          }

        }
        output {
          leaf-list resource-list {
            description
              "Provides the set of Resources associated with the profile
              provided.";

            type leafref {
              path "/rpm:performance-monitoring/rpm:resources" +
              "/rpm:resource-list/rpm:resource";
            }
          }
        }
```

```
    }

    grouping pm-data-instance-grouping {
      description
        "grouping for common attributes of performance data";

      leaf resource {
      description
        "the identifier of network resource which is monitored.";

        type leafref {
          path "/rpm:performance-monitoring/rpm:resources" +
          "/rpm:resource-list/rpm:resource";
        }
      }

      leaf-list task-id {
         description
          "the task id list of the tasks from which the pm data is retrieved";

          type yang:uuid;
      }
      leaf collect-time {
        description
          "the time of this data is collected";

        type yang:date-and-time;
      }

      leaf resource-type {
        description
          "the type of resource, such as NE, board or port";

        type identityref {
          base rpm-types:resource-type;
        }
      }

      container indicator-data {
        description
          "grouping for historic performance data";

        list indicator-data-list {
          description
            "list of historic performance data";
          key indicator-name;

          uses indicator-data-instance-grouping;
```

```
        }
      }
    }

    grouping indicator-data-instance-grouping {
      description
        "grouping for a performance data";

      leaf indicator-name {
        description
          "name of performance data indicator";

        type string;
      }

      leaf indicator-value {
        description
          "value of performance data";

        type string;
      }

      leaf indicator-value-unit {
        description
          "unit of performance data value";

        type string;
      }
    }
  }
  <CODE ENDS>
```

Figure 2: Resource Performance Monitoring YANG module

5. YANG Model for Resource Performance Monitoring Types

```
  <CODE BEGINS> file "ietf-resource-pm-types@2025-07-04.yang"
  module ietf-resource-pm-types {
    yang-version 1.1;
    namespace
    "urn:ietf:params:xml:ns:yang:ietf-resource-pm-types";

    prefix rpm-types;

    organization
      "IETF CCAMP Working Group";
    contact
      "WG Web:   <https://datatracker.ietf.org/wg/ccamp/>
```

```
        WG List:  <mailto:ccamp@ietf.org>

        Editor:   Chaode Yu
                  <yuchaode@huawei.com>

        Editor:   Fabio Peruzzini
                  <fabio.peruzzini@fibercop.com>

        Editor:   Yanlei Zheng
                  <zhengyanlei@chinaunicom.cn>

        Editor:   Victor Lopez
                  <victor.lopez@nokia.com>

        Editor:   Italo Busi
                  <italo.busi@huawei.com>

        Editor:   Aihua Guo
                  <aihuaguo.ietf@gmail.com>

        Editor:   Xing Zhao
                  <zhaoxing@caict.ac.cn>

        Editor:   Mingshuang Jin
                  <jinmingshuang@huawei.com>";

     description
       "This module defines types model for resource performance
       monitoring which will be imported by ietf-resource-pm
       data model.

       The model fully conforms to the Network Management
       Datastore Architecture (NMDA).

       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).

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

       The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL
```

```
       NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'NOT RECOMMENDED',
       'MAY', and 'OPTIONAL' in this document are to be interpreted as
       described in BCP 14 (RFC 2119) (RFC 8174) when, and only when,
       they appear in all capitals, as shown here.";

     revision 2025-07-04 {
       description  "Initial revision.";

     }

     identity resource-type {
       description "this is the base type of all the rerource type";
     }

     identity network-element {
       base resource-type;
       description "NE resource type";
     }

      identity interface {
         base resource-type;
         description "Network interface";
      }

     identity board {
       base resource-type;
       description "board resource type";
     }

     identity termination-point {
       base resource-type;
       description "Termination point resource";
     }

     identity tunnel-termination-point {
       base resource-type;
       description "Tunnel termination point resource";
     }

     identity period {
       description
         "this is the base type of all the performace monitoring priod
         type.";
     }

     identity period-15-minutes {
       base period;
       description
```

```
         "the during of monitoring task will be repeated at every 15
         minutes";
     }

     identity period-24-hours {
       base period;
       description
          "the during of monitoring task will be repeated at every 24
          hours";
     }

     identity severity {
       description
           "it is used to indicate what severity alarm will be caused if
           exceeds the threshold";
     }

     identity critical {
       description
           "critical alarm will be caused if exceeds the threshold";
       base severity;
     }

     identity major {
       description
           "major alarm will be caused if exceeds the threshold";
       base severity;
     }

     identity minor {
       description
           "minor alarm will be caused if exceeds the threshold";
       base severity;
     }

     identity warning {
       description
           "only a warning will be caused if exceeds the threshold";
       base severity;
     }

     identity layer-rate-type {
       description
           "It is used to indicate the layer rate of network element when
           retrieving the pm parameters supported";
     }
   }
   <CODE ENDS>
```

Figure 3: Resource Performance Monitoring Types YANG module

6. Manageability Considerations

<Add any manageability considerations>

7. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The NETCONF access control model [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). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. Considerations in Section 8 of [RFC8795]are also applicable to their subtrees in the module defined in this document.

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. Considerations in Section 8 of [RFC8795] are also applicable to their subtrees in the module defined in this document.

8. IANA Considerations

This document registers following YANG modules in the YANG Module Names registry [RFC6020].

name: ietf-resource-pm namespace: urn:ietf:params:xml:ns:yang:ietf-resource-pm prefix: dvcrpm reference: RFC XXXX: A YANG Data Model for Resource Performance Monitoring

name: ietf-resource-pm-types namespace: urn:ietf:params:xml:ns:yang:ietf-resource-pm-types prefix: dvcrpm-types reference: RFC XXXX: A YANG Data Model for Resource Performance Monitoring

9. References

9.1. Normative References

[ITU-T_G.874] International Telecommunication Union, "Management aspects of optical transport network elements", ITU-T Recommendation G.874 , October 2020, <https://www.itu.int/rec/T-REC-G.874/en>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/rfc/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/rfc/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/rfc/rfc6242>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/rfc/rfc6991>.

[RFC7446] Lee, Y., Ed., Bernstein, G., Ed., Li, D., and W. Imajuku, "Routing and Wavelength Assignment Information Model for Wavelength Switched Optical Networks", RFC 7446, DOI 10.17487/RFC7446, February 2015, <https://www.rfc-editor.org/rfc/rfc7446>.

[RFC7581] Bernstein, G., Ed., Lee, Y., Ed., Li, D., Imajuku, W., and J. Han, "Routing and Wavelength Assignment Information Encoding for Wavelength Switched Optical Networks", RFC 7581, DOI 10.17487/RFC7581, June 2015, <https://www.rfc-editor.org/rfc/rfc7581>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/rfc/rfc7950>.

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

[RFC8194] Schoenwaelder, J. and V. Bajpai, "A YANG Data Model for LMAP Measurement Agents", RFC 8194, DOI 10.17487/RFC8194, August 2017, <https://www.rfc-editor.org/rfc/rfc8194>.

[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/rfc/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/rfc/rfc8341>.

[RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <https://www.rfc-editor.org/rfc/rfc8342>.

[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/rfc/rfc8446>.

[RFC8795] Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Gonzalez de Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", RFC 8795, DOI 10.17487/RFC8795, August 2020, <https://www.rfc-editor.org/rfc/rfc8795>.

[RFC9375] Wu, B., Ed., Wu, Q., Ed., Boucadair, M., Ed., Gonzalez de Dios, O., and B. Wen, "A YANG Data Model for Network and VPN Service Performance Monitoring", RFC 9375, DOI 10.17487/RFC9375, April 2023, <https://www.rfc-editor.org/rfc/rfc9375>.

[TMF-518] TM Forum (TMF), "Resource Performance Management", TMF518_RPM , 2011, <https://www.tmforum.org/resources/collection/mtosi-4-0>.

9.2. Informative References

[I-D.ietf-opsawg-yang-vpn-service-pm] Wu, B., Wu, Q., Boucadair, M., de Dios, O. G., and B. Wen, "A YANG Data Model for Network and VPN Service Performance Monitoring", Work in Progress, Internet-Draft, draft-ietf-

opsawg-yang-vpn-service-pm-15, 11 November 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-opsawg-yang-vpn-service-pm-15>.

[I-D.ietf-teas-actn-pm-telemetry-autonomics] Lee, Y., Dhody, D., Vilalta, R., King, D., and D. Ceccarelli, "YANG models for Virtual Network (VN)/TE Performance Monitoring Telemetry and Scaling Intent Autonomics", Work in Progress, Internet-Draft, draft-ietf-teas-actn-pm-telemetry-autonomics-16, 13 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-teas-actn-pm-telemetry-autonomics-16>.

[I-D.zheng-ccamp-client-pm-yang] Yu, C., Zheng, H., Busi, I., Yanlei, Z., Lopez, V., and O. G. de Dios, "A YANG Data Model for Client Signal Performance Monitoring", Work in Progress, Internet-Draft, draft-zheng-ccamp-client-pm-yang-13, 29 August 2025, <https://datatracker.ietf.org/doc/html/draft-zheng-ccamp-client-pm-yang-13>.

[RFC8454] Lee, Y., Belotti, S., Dhody, D., Ceccarelli, D., and B. Yoon, "Information Model for Abstraction and Control of TE Networks (ACTN)", RFC 8454, DOI 10.17487/RFC8454, September 2018, <https://www.rfc-editor.org/rfc/rfc8454>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/rfc/rfc8641>.

Acknowledgments

This document was prepared using kramdown.

Authors' Addresses

Chaode Yu
Huawei Technologies
Email: yuchaode@huawei.com

Fabio Peruzzini
FiberCop
Email: fabio.peruzzini@fibercop.com

Yanlei Zheng
China Unicom

Email: zhengyanlei@chinaunicom.cn

Italo Busi
Huawei Technologies
Email: italo.busi@huawei.com

Aihua Guo
Futurewei Technologies
Email: aihuaguo.ietf@gmail.com

Victor Lopez
Nokia
Email: victor.lopez@nokia.com

Xing Zhao
CAICT
Email: zhaoxing@caict.ac.cn

Mingshuang Jin
Huawei Technologies
Email: jinmingshuang@huawei.com