TEAS WG
Internet Draft
Intended Status: standard

Young Lee
Dhruv Dhody
Satish Karunanithi
Huawei
Ricard Vilalta
CTTC
Daniel King
Lancaster University
Daniele Ceccarelli
Ericsson

Expires: January 2, 2018

July 3, 2017

YANG models for ACTN TE Performance Monitoring Telemetry and Network Autonomics

draft-lee-teas-actn-pm-telemetry-autonomics-02

Status of this Memo

Abstract

Abstraction and Control of TE Networks (ACTN) refers to the set of virtual network operations needed to operate, control and manage large-scale multi-domain, multi-layer and multi-vendor TE networks, so as to facilitate network programmability, automation, efficient resource sharing.

This document provides YANG data models that describe Key Performance Indicator (KPI) telemetry and network autonomics for TE-tunnels and ACTN VNs.

Table of Contents

1. Introduction

Abstraction and Control of TE Networks (ACTN) describes a method for operating a Traffic Engineered (TE) network (such as an MPLS-TE network or a layer 1/0 transport network) to provide connectivity and virtual network services for customers of the TE network [ACTN-Frame]. The services provided can be optimized to meet the requirements (such as traffic patterns, quality, and reliability) of the applications hosted by the customers. Data models are a representation of objects that can be configured or monitored within a system. Within the IETF, YANG [RFC6020] is the language of choice for documenting data models, and YANG models have been produced to allow configuration or modeling of a variety of network devices, protocol instances, and network services. YANG data models have been classified in [Netmod-Yang-Model-Classification] and [Service-YANG].

[ACTN-VN-YANG] describes how customers or end to end orchestrators can request and/or instantiate a generic virtual network service. [ACTN-Applicability] describes a connection between IETF YANG model classifications to ACTN interfaces. In particular, it describes the customer service model can be mapped into the CMI (CNC-MDSC Interface) of the ACTN architecture.

The YANG model on the ACTN CMI is known as customer service model in [Service-YANG]. [PCEP-Service-Aware] describes key network performance data to be considered for end-to-end path computation in TE networks. Key performance indicator is a term that describes critical performance data that may affect VN/TE service.

2. Use-Cases

[ACTN-PERF] describes use-cases relevant to this draft. It introduces the dynamic creation, modification and optimization of services based on the performance monitoring in the Abstraction and Control of Transport Networks (ACTN) architecture. Figure 1 shows a high-level workflows for dynamic service control based on traffic monitoring.

Some of the key points from [ACTN-PERF] are as follows:

- Network traffic monitoring is important to facilitate automatic discovery of the imbalance of network traffic, and initiate the network optimization, thus helping the network operator or the virtual network service provider to use the network more efficiently and save CAPEX/OPEX.
- Customer services have various SLA requirements, such as service availability, latency, latency jitter, packet loss rate, BER, etc. The transport network can satisfy service availability and BER requirements by providing different protection and restoration mechanisms. However, for other performance parameters, there are no such mechanisms. In order to provide high quality services according to customer SLA, one possible solution is to measure the service SLA related performance parameters, and dynamically provision and optimize services based on the performance monitoring results.
- Performance monitoring in a large scale network could generate a huge amount of performance information. Therefore, the appropriate way to deliver the information in CMI and MPI interfaces should be carefully considered.

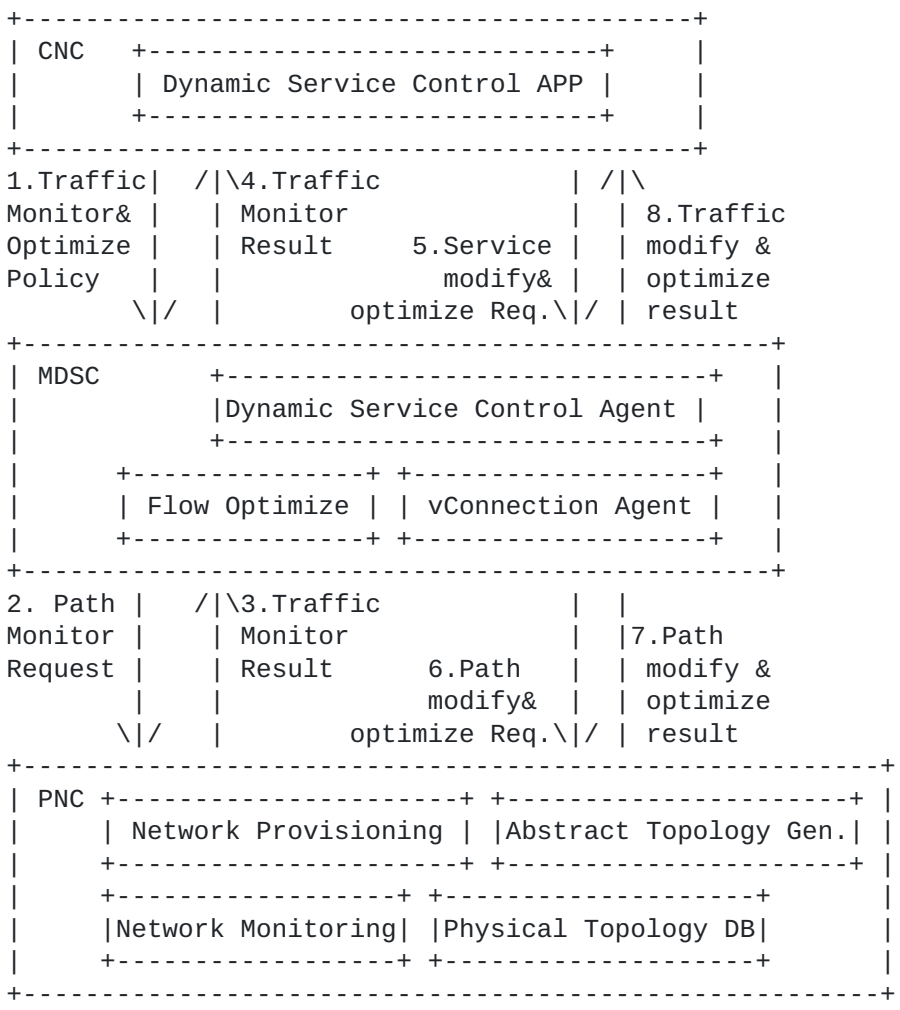


Figure 1 Workflows for dynamic service control based on traffic monitoring

3. Design of the Data Models

The YANG models developed in this document describe two models:

(i) TE KPI Telemetry Model which provides the TE-Tunnel level of performance monitoring mechanism (See Section 2.1 for details)

(ii) ACTN TE KPI Telemetry Model which provides the VN level of the aggregated performance monitoring mechanism (See Section 2.2 for details)

The models include -

(i) Performance Telemetry details as measured during the last interval, ex delay.

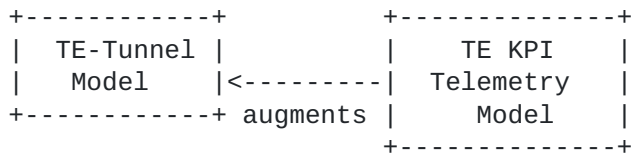
(ii) Scaling Intent based on with TE/VN could be scaled in/out.

[Editor's Note - Need to decide if scaling and telemetry can be in the same model as per the current draft.]

TE KPI Telemetry Model

This module describes performance telemetry for TE-tunnel model. The telemetry data is augmented to tunnel state. This module also allows autonomic traffic engineering scaling intent configuration mechanism on the TE-tunnel level. Various conditions can be set for auto-scaling based on the telemetry data.

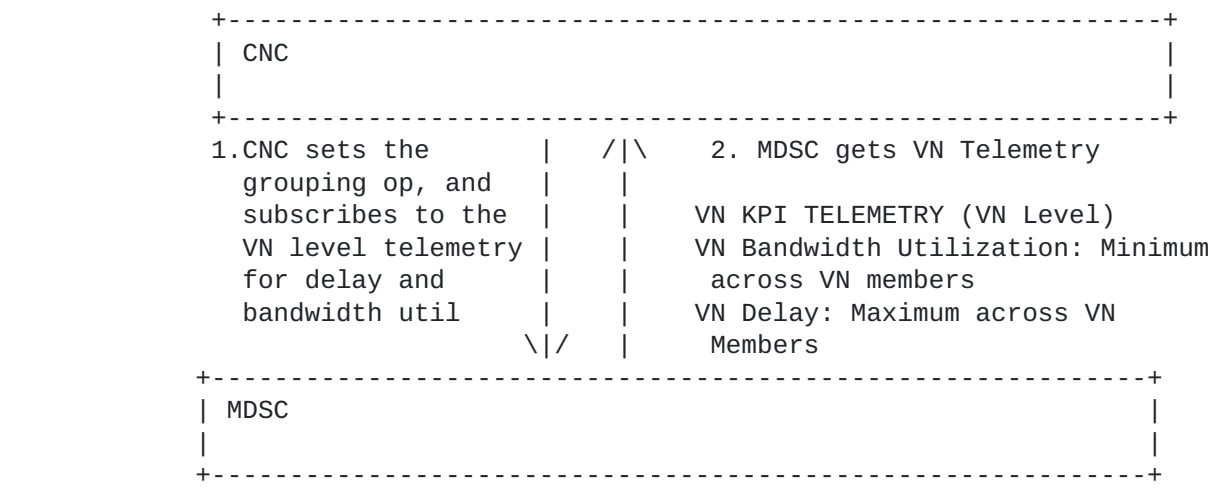
The TE KPI Telemetry Model augments the TE-Tunnel Model to enhance TE performance monitoring capability. This monitoring capability will facilitate proactive re-optimization and reconfiguration of TEs based on the performance monitoring data collected via the TE KPI Telemetry YANG model.



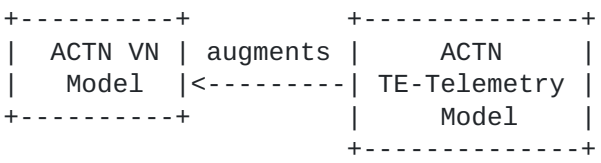
ACTN TE KPI Telemetry Model

This module describes performance telemetry for ACTN VN model. The telemetry data is augmented both at the VN Level as well as individual VN member level. This module also allows autonomic traffic engineering scaling intent configuration mechanism on the VN level. Scale in/out criteria might be used for network autonomics in order the controller to react to a certain set of variations in monitored parameters.

Moreover, this module also provides mechanism to define aggregated telemetry parameters as a grouping of underlying VN level telemetry parameters. Grouping operation (such as maximum, mean) could be set at the time of configuration. For example, if maximum grouping operation is used for delay at the VN level, the VN telemetry data is reported as the maximum {delay_vn_member_1, delay_vn_member_2, .. delay_vn_member_N}. Thus, this telemetry abstraction mechanism allows the grouping of a certain common set of telemetry values under a grouping operation. This can be done at the VN-member level to suggest how the E2E telemetry be inferred from the per domain tunnel created and monitored by PNCs. One proposed example is the following:



The ACTN VN TE-Telemetry Model augments the basic ACTN VN model to enhance VN monitoring capability. This monitoring capability will facilitate proactive re-optimization and reconfiguration of VNs based on the performance monitoring data collected via the ACTN VN Telemetry YANG model.



4. Notification

This model does not define specific notifications. To enable notifications, the mechanism defined in [I-D.ietf-netconf-yang-push] and [I-D.ietf-netconf-rfc5277bis] can be used. This mechanism currently allows the user to:

. Subscribe notifications on a per client basis.

. Specify subtree filters or xpath filters so that only interested contents will be sent.

. Specify either periodic or on-demand notifications.

YANG Push Subscription Examples

Below example shows the way for a client to subscribe for the telemetry information for a particular tunnel (Tunnel1). The telemetry parameter that the client is interested in is the utilized bandwidth.

```
<netconf:rpc netconf:message-id="101"
    xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <establish-subscription
       xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push:1.0">
       <filter netconf:type="subtree">
          <te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
             <tunnels>
                <tunnel>
                  <name>Tunnel1</name>
                  <identifier/>
                  <state>
                    <te-telemetry
xmlns="urn:ietf:params:xml:ns:yang:ietf-te-kpi-telemetry">
```

```
                          <utilized-
   bandwidth/>

                       </te-telemetry>
                     </state>
                    </tunnel>
                 </tunnels>
              </te>
           </filter>
           <period>500</period>
           <encoding>encode-xml</encoding>
       </establish-subscription>
    </netconf:rpc>
```

This example shows the way for a client to subscribe for the telemetry information for all VNs. The telemetry parameter that the client is interested in is packet-loss and utilized bandwidth.

```
<netconf:rpc netconf:message-id="101"
    xmlns:netconf="urn:ietf:params:xml:ns:netconf:base:1.0">
    <establish-subscription
       xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push:1.0">
       <filter netconf:type="subtree">
          <actn-state xmlns="urn:ietf:params:xml:ns:yang:ietf-actn-
vn">
             <vn>
                <vn-list>
                  <vn-id/>
                  <vn-name/>
                  <vn-
telemetry  xmlns="urn:ietf:params:xml:ns:yang:ietf-actn-te-kpi-
telemetry">
                      <one-way-packet-loss/>
                      <utilized-
bandwidth/>

                  </vn-telemetry >
                </vn-list>
             </vn>
          </actn-state>
       </filter>
       <period>500</period>
    </establish-subscription>
 </netconf:rpc>
```

5. YANG Data Tree

A graphical representation of the complete data tree is presented here. The meaning of the symbols in these diagrams is as follows and as per [I-D.ietf-netmod-rfc6087bis]. Each node is printed as: <status> <flags> <name> <opts> <type> <if-features>

<status> is one of:
  + for current
  x for deprecated
  o for obsolete

<flags> is one of:
  rw for configuration data
  ro for non-configuration data
  -x for rpcs and actions
  -n for notifications

<name> is the name of the node
  (<name>) means that the node is a choice node
 :(<name>) means that the node is a case node

If the node is augmented into the tree from another module, its name is printed as <prefix>:<name>.

<opts> is one of:
  ? for an optional leaf, choice, anydata or anyxml
  ! for a presence container
  * for a leaf-list or list
  [<keys>] for a list's keys

<type> is the name of the type for leafs and leaf-lists

  If the type is a leafref, the type is printed as "-> TARGET",

  where TARGET is either the leafref path, with prefixed removed if possible.

<if-features> is the list of features this node depends on,

printed within curly brackets and a question mark "{...}?

```
module: ietf-te-kpi-telemetry
  augment /te:te/te:tunnels/te:tunnel/te:config:
    +--rw te-scaling-intent
       +--rw scale-in
       |  +--rw scale-in-operation-type?
       |  |       scaling-criteria-operation
       |  +--rw threshold-time?            uint32
       |  +--rw scale-in-condition* [performance-type]
       |     +--rw performance-type    identityref
       |     +--rw performance-data?   binary
       +--rw scale-down
          +--rw cooldown-time?              uint32
          +--rw scale-out-operation-type?
          |       scaling-criteria-operation
          +--rw scale-out-condition* [performance-type]
             +--rw performance-type    identityref
             +--rw performance-data?   binary
  augment /te:te/te:tunnels/te:tunnel/te:state:
    +--ro te-telemetry
       +--ro data
          +--ro one-way-delay?             uint32
          +--ro two-way-delay?             uint32
          +--ro one-way-delay-min?         uint32
          +--ro one-way-delay-max?         uint32
          +--ro two-way-delay-min?         uint32
          +--ro two-way-delay-max?         uint32
          +--ro one-way-delay-variation?   uint32
          +--ro two-way-delay-variation?   uint32
          +--ro utilized-bandwidth?        tet:te-bandwidth
```

```
module: ietf-actn-te-kpi-telemetry
  augment /actn-vn:actn/actn-vn:vn/actn-vn:vn-list:
    +--rw vn-telemetry
    |  +--rw grouping-op
    |  |  +--rw delay-op?               identityref
    |  |  +--rw delay-variation-op?     identityref
    |  |  +--rw utilized-bandwidth-op?  identityref
    |  +--ro data
    |     +--ro one-way-delay?            uint32
    |     +--ro two-way-delay?            uint32
    |     +--ro one-way-delay-min?        uint32
    |     +--ro one-way-delay-max?        uint32
```

```
    |     +--ro two-way-delay-min?         uint32
    |     +--ro two-way-delay-max?         uint32
    |     +--ro one-way-delay-variation?   uint32
    |     +--ro two-way-delay-variation?   uint32
    |     +--ro utilized-bandwidth?        tet:te-bandwidth
    +--rw vn-scaling-intent
       +--rw scale-in
       |  +--rw scale-in-operation-type?
       |  |       scaling-criteria-operation
       |  +--rw threshold-time?            uint32
       |  +--rw scale-in-condition* [performance-type]
       |     +--rw performance-type    identityref
       |     +--rw performance-data?   binary
       +--rw scale-down
          +--rw cooldown-time?              uint32
          +--rw scale-out-operation-type?
          |       scaling-criteria-operation
          +--rw scale-out-condition* [performance-type]
             +--rw performance-type    identityref
             +--rw performance-data?   binary
  augment /actn-vn:actn/actn-vn:vn/actn-vn:vn-list/actn-vn:vn-member-list:
    +--rw vn-telemetry
       +--rw grouping-op
       |  +--rw delay-op?                identityref
       |  +--rw delay-variation-op?      identityref
       |  +--rw utilized-bandwidth-op?   identityref
       +--ro data
          +--ro one-way-delay?             uint32
          +--ro two-way-delay?             uint32
          +--ro one-way-delay-min?         uint32
          +--ro one-way-delay-max?         uint32
          +--ro two-way-delay-min?         uint32
          +--ro two-way-delay-max?         uint32
          +--ro one-way-delay-variation?   uint32
          +--ro two-way-delay-variation?   uint32
          +--ro utilized-bandwidth?        tet:te-bandwidth
```

6. Yang Data Model

ietf-te-kpi-telemetry model

The YANG code is as follows:

<CODE BEGINS> file "ietf-te-kpi-telemetry@2017-07-03.yang"

```
module ietf-te-kpi-telemetry {
  namespace "urn:ietf:params:xml:ns:yang:ietf-te-kpi-telemetry";

    prefix "te-tel";

    import ietf-te {
        prefix "te";
    }

    import ietf-te-topology {
       prefix "tet";
    }

    organization
        "IETF Traffic Engineering Architecture and Signaling (TEAS)
        Working Group";

    contact
        "Editor: Young Lee <leeyoung@huawei.com>
         Editor: Dhruv Dhody <dhruv.ietf@gmail.com>
         Editor: Ricard Vilalta <ricard.vilalta@cttc.es>
         Editor: Satish Karunanithi <satish.karunanithi@gmail.com>";
    description
        "This module describes telemetry for teas tunnel model";

    revision 2017-07-03 {
        description
            "Initial revision. This YANG file defines
            the reusable base types for TE telemetry.";
        reference
            "xxx";
    }

    /*
     * Identities
     */

    identity telemetry-param-type {
        description
            "Base identity for telemetry param types";
    }

    identity one-way-delay {
        base telemetry-param-type;
        description
            "To specify average Delay in one (forward) direction";
    }
```

```
    identity two-way-delay {
        base telemetry-param-type;
        description
            "To specify average Delay in both (forward and reverse)
             directions";
    }

    identity one-way-delay-variation {
        base telemetry-param-type;
        description
            "To specify average Delay Variation in one (forward)
            direction";
    }

    identity two-way-delay-variation {
        base telemetry-param-type;
        description
            "To specify average Delay Variation in both (forward
            and reverse) directions";
    }

    identity utilized-bandwidth {
        base telemetry-param-type;
        description
            "To specify utilized bandwidth over the specified source
             and destination.";
    }

    /*
     * Enums
     */
    typedef scaling-criteria-operation {
        type enumeration {
            enum AND {
                description
                 "AND operation";
            }
            enum OR {
              description
               "OR operation";
            }
        }
        description
         "Operations to analize list of scaling criterias";
    }

    /*
```

```
     * Groupings
     */

    grouping telemetry-delay {
      description
      "Base telemetry delay parameters";

      leaf one-way-delay {
            type uint32;
            units "microseconds";
      description
          "To specify average Delay in one (forward) direction
               during the measurement interval";
      }
      leaf two-way-delay {
            type uint32;
            units "microseconds";
      description
          "To specify average Delay in both (forward and reverse)
              directions during the measurement interval";
      }

      leaf one-way-delay-min {
            type uint32;
            units "microseconds";
      description
          "To specify minimum Delay in one (forward) direction
           during the measurement interval";
      }
      leaf one-way-delay-max {
            type uint32;
            units "microseconds";
      description
          "To specify maximum Delay in one (forward) direction
           during the measurement interval";
      }

      leaf two-way-delay-min {
            type uint32;
            units "microseconds";
      description
          "To specify minimum Delay in both (forward and reverse)
              directions during the measurement interval";
      }

      leaf two-way-delay-max {
            type uint32;
```

```
            units "microseconds";
      description
          "To specify maximum Delay in both (forward and reverse)
               directions during the measurement interval";
      }

    }

    grouping telemetry-delay-variance {

    description
      "Base telemetry delay variance parameters";
    leaf one-way-delay-variation {
      type uint32;
      units "microseconds";
      description
        "To specify average Delay Variation in one (forward)
        direction during the measurement interval";
    }

    leaf two-way-delay-variation {
      type uint32;
      units "microseconds";
      description
        "To specify average Delay Variation in both
        (forward and reverse) directions during the
        measurement interval";
    }
  }

    grouping telemetry-bandwidth {
      description
          "Base telemetry bandwidth parameters";
        leaf utilized-bandwidth {
            type tet:te-bandwidth;
        description
          "To specify utilized bandwidth over the specified source
            and destination in bytes per seconds.";
            reference
                "RFC 3471";
        }
    }

    grouping scaling-criteria {
      description
            "Grouping for scaling criteria";
        leaf performance-type {
```

```
          type identityref {
              base telemetry-param-type;
            }
            description
              "Reference to the tunnel level telemetry type";
        }

        leaf performance-data {
          type binary;
          description
              "The encoding and meaning of this field is
              based on the performance-type";
        }
    }
    grouping scaling-intent {
      description
        "Basic scaling intent";

        container scale-in {
            description
            "Basic scaling-in intent";

            leaf scale-in-operation-type {
                type scaling-criteria-operation;
                default AND;
                description
                    "Operation to be applied to check between
                    scaling criterias to check if the scale in
                    threshold condition has been met.
                    Defaults to AND";
            }

            leaf threshold-time {
                type uint32;
                units "seconds";
                description
                    "The duration for which the criteria must
                    hold true";
            }

            list scale-in-condition {
              key "performance-type";
              description
                "Scaling conditions";
              uses scaling-criteria;
          }
        }
```

```
        container scale-down {
            description
            "Basic scaling-out intent";
            leaf cooldown-time {
                type uint32;
                units "seconds";
                description
                "The duration after a scaling-in/scaling-out action
                has been triggered, for which there will be no
                further operation";
            }
            leaf scale-out-operation-type {
                type scaling-criteria-operation;
                default OR;
                description
                    "Operation to be applied to check between
                    scaling criterias to check if the scale out
                    threshold condition has been met.
                    Defauls to OR";
            }
            list scale-out-condition {
              key "performance-type";
              description
                "Scaling conditions";
              uses scaling-criteria;
          }

        }

    }

    grouping telemetry-param {

        description
      "Base telemetry parameters";
    container data {
      config false;
      description
        "The telemetry data";

          uses telemetry-delay;

          uses telemetry-delay-variance;

          uses telemetry-bandwidth;
      }
```

```
    }

    /*
     * Augments
     */
    augment "/te:te/te:tunnels/te:tunnel/te:config" {

        description
            "Augmentation parameters for config scaling-criteria
            TE tunnel topologies. Scale in/out criteria might be
            used for network autonomics in order the controller
            to react to a certain set of monitored params.";

        container te-scaling-intent {
          description
            "scaling intent";
            uses scaling-intent;

        }

    }

    augment "/te:te/te:tunnels/te:tunnel/te:state" {

        description
            "Augmentation parameters for state TE tunnel
            topologies.";

        container te-telemetry {
          description
            "telemetry params";
          uses telemetry-param;
        }
    }

}//module

<CODE ENDS>
```

ietf-actn-te-kpi-telemetry model

The YANG code is as follows:

```
<CODE BEGINS> file "ietf-actn-te-kpi-telemetry@2017-07-03.yang"
```

```
module ietf-actn-te-kpi-telemetry {

    namespace
      "urn:ietf:params:xml:ns:yang:ietf-actn-te-kpi-telemetry";

    prefix "actn-tel";

    import ietf-actn-vn {
        prefix "actn-vn";
    }

    import ietf-te-kpi-telemetry {
        prefix "te-kpi";
    }

    organization
        "IETF Traffic Engineering Architecture and Signaling (TEAS)
        Working Group";

    contact
        "Editor: Young Lee <leeyoung@huawei.com>
         Editor: Dhruv Dhody <dhruv.ietf@gmail.com>
         Editor: Ricard Vilalta <ricard.vilalta@cttc.es>
         Editor: Satish Karunanithi <satish.karunanithi@gmail.com>";

    description
        "This module describes telemetry for actn vn model";

    revision 2017-07-03 {
        description
            "Initial revision. This YANG file defines
            the reusable base types for ACTN VN telemetry.";
        reference
            "xxx";
    }

    /*
     * Identities
     */

    identity grouping-operation {
        description "Base identity for operations to analize list of monitored
param";
    }

    identity minimum-grouping-operation {
        base grouping-operation;
        description
            "Select the minimum param";
    }
```

```
    identity maximum-grouping-operation {
        base grouping-operation;
        description
            "Select the maximum param";
    }

    identity mean-grouping-operation {
        base grouping-operation;
        description
            "Select the MEAN of the params";
    }

    identity stddev-grouping-operation {
        base grouping-operation;
        description
            "Select the STD deviation of the params";
    }

    identity sum-grouping-operation {
        base grouping-operation;
        description
            "Select the sum of the params";
       reference
            "RFC 7823";
    }

    /*
     * Groupings
     */
    grouping vn-telemetry-param {
        description
            "telemetry-parameter for VN";
        uses te-kpi:telemetry-param;
    }
    grouping telemetry-grouping-op {
        description
            "Config how the VN telemetry should be applied";
        container grouping-op {
            description
                "The grouping operations";
            leaf delay-op {
                type identityref {
                  base grouping-operation;
                }
                default maximum-grouping-operation;
                description
                    "The operation that should be applied on the
                     VN-member telemetry to get the VN telemetry";
```

```
            }

            leaf delay-variation-op {
                type identityref {
                  base grouping-operation;
                }
                default maximum-grouping-operation;
                description
                    "The operation that should be applied on the
                     VN-member telemetry to get the VN telemetry";
            }

            leaf utilized-bandwidth-op {
                type identityref {
                  base grouping-operation;
                }
                default maximum-grouping-operation;
                description
                    "The operation that should be applied on the
                     VN-member telemetry to get the VN telemetry";
            }
        }

    }

    /*
     * Augments
     */
    augment "/actn-vn:actn/actn-vn:vn/actn-vn:vn-list" {

        description
            "Augmentation parameters for state TE VN topologies.";

        container vn-telemetry {
          description
            "VN telemetry configurations";
            uses telemetry-grouping-op;
            uses vn-telemetry-param;
        }
        container vn-scaling-intent {
            description
                "scaling intent";
            uses te-kpi:scaling-intent;
        }
    }

    /*
     * VN-member augment
     */
```

```
    augment "/actn-vn:actn/actn-vn:vn/actn-vn:vn-list/" +
            "actn-vn:vn-member-list" {
        description
            "Augmentation parameters for state TE vn member
            topologies.";
        container vn-telemetry {
            description
                "VN Member config";
            uses telemetry-grouping-op;
            uses vn-telemetry-param;
        }
    }
}<CODE ENDS>
```

7. Security Considerations

The configuration, state, and action data defined in this document are designed to be accessed via a management protocol with a secure transport layer, such as NETCONF [RFC6241]. The NETCONF access control model [RFC6536] provides the means to restrict access for particular NETCONF users to a preconfigured subset of all available NETCONF protocol operations and content.

A number of configuration data nodes defined in this document are writable/deletable (i.e., "config true") These data nodes may be considered sensitive or vulnerable in some network environments.

8. IANA Considerations

TDB

9. Acknowledgements

10. References

Informative References

[RFC4110] R. Callon and M. Suzuki, "A Framework for Layer 3 Provider-Provisioned Virtual Private Networks (PPVPNs)", RFC 4110, July 2005.

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

[Service-YANG] Q. Wu, W. Liu and A. Farrel, "Service Models Explained", draft-wu-opsawg-service-model-explained, work in progress.

[Netmod-Yang-Model-Classification] D. Bogdanovic, B. Claise, and C. Moberg, "YANG Module Classification", draft-ietf-netmod-yang-model-classification, work in progress.

[Netconf] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241.

[Restconf] A. Bierman, M. Bjorklund, and K. Watsen, "RESTCONF Protocol", draft-ietf-netconf-restconf, work in progress.

Normative References

[ACTN-Frame] D. Cecarelli and Y. Lee, "Framework for Abstraction and Control of Traffic Engineered Networks", draft-ietf-teas-actn-framework, work in progress.

[TE-Topology] X. Liu, et al., "YANG Data Model for TE Topologies", draft-ietf-teas-yang-te-topo, work in progress.

[TE-Tunnel] T. Saad (Editor), "A YANG Data Model for Traffic Engineering Tunnels and Interfaces", draft-ietf-teas-yang-te, work in progress.

[ACTN-VN-YANG] Y. Lee (Editor), "A Yang Data Model for ACTN VN Operation", draft-lee-teas-actn-vn-yang, work in progress.

[L3SM-YANG] S. Litkowski, L.Tomotaki, and K. Ogaki, "YANG Data Model for L3VPN service delivery", draft-ietf-l3sm-l3vpn-service-model, work in progress.

[PCEP-Service-Aware] D. Dhody, et al., "Extensions to the Path Computation Element Communication Protocol (PCEP) to compute service aware Label Switched Path (LSP)", draft-ietf-pce-pcep-service-aware, work in progress.

[ACTN-PERF] Y. XU, et al., "Use Cases and Requirements of Dynamic Service Control based on Performance Monitoring in ACTN Architecture", draft-xu-actn-perf-dynamic-service-control-03, work in progress.

11. Contributors

Authors' Addresses

Young Lee
Huawei Technologies
5340 Legacy Drive Suite 173
Plano, TX 75024, USA

Email: leeyoung@huawei.com

Dhruv Dhody
Huawei Technology
Leela Palace
Bangalore, Karnataka 560008
India

Email: dhruv.dhody@huawei.com

Satish Karunanithi
Huawei Technology
Leela Palace
Bangalore, Karnataka 560008
India

Email: satish.karunanithi@gmail.com

Ricard Vilalta
Centre Tecnologic de Telecomunicacions de Catalunya (CTTC/CERCA)
Av. Carl Friedrich Gauss 7
08860 - Castelldefels
Barcelona (Spain)
Email: ricard.vilalta@cttc.es

Daniel King
Lancaster University

Email: d.king@lancaster.ac.uk

Daniele Ceccarelli
Ericsson
Torshamnsgatan,48
Stockholm, Sweden

Email: daniele.ceccarelli@ericsson.com