Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 10, 2015

A. Clemm
J. Medved
Cisco
R. Varga
T. Tkacik
Pantheon Technologies SRO
X. Liu
Ericsson
I. Bryskin
A. Guo
Adva Optical
H. Ananthakrishnan
Packet Design
N. Bahadur
Bracket Computing
V. Beeram
Juniper Networks

June 8, 2015

A YANG Data Model for Layer 3 Topologies
draft-ietf-i2rs-yang-l3-topology-00.txt

Abstract

This document defines a YANG data model for layer 3 network topologies.

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 http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 10, 2015.

Table of Contents

1. Introduction

This document introduces a YANG [RFC6020] [RFC6021] data model for Layer 3 network topologies. The model allows an application to have a holistic view of the topology of a Layer 3 network, all contained in a single conceptual YANG datastore. The data model builds on top of, and augments, the data model for network topologies defined in [I-D.draft-ietf-i2rs-yang-network-topo]. An earlier revision of that Internet Draft contained not just the general model for network topologies, but also the model for layer 3 network topologies that is being specified here. However, we decided to "split" the earlier draft to separate the truly general aspects of a topology data model, which apply to any type of topology, from the application of this model to a particular domain, here: a Layer 3 network.

Specific topology types that are covered in this document include Layer 3 Unicast IGP, IS-IS [RFC1195], and OSPF [RFC2178]. In addition, this documents defines a set of traffic engineering extensions.

There are multiple applications for such a data model and a number of use cases have been defined in section 6 of [I-D.draft-ietf-i2rs-usecase-reqs-summary]. For example, nodes within the network can use the data model to capture their understanding of the overall network topology and expose it to a network controller. A network controller can then use the instantiated topology data to compare and reconcile its own view of the network topology with that of the network elements that it controls. Alternatively, nodes within the network could propagate this understanding to compare and reconcile this understanding either amongst themselves or with help of a controller. Beyond the network element itself, a network controller might even use the data model to represent its view of the topology that it controls and expose it to applications north of itself.

There are several reasons to choose YANG to define the data model. Data defined using YANG can be exposed by a server to client applications and controllers via Netconf [RFC6241] or via a ReST Interface [I-D.draft-ietf-netconf-restconf] [I-D.draft-ietf-netmod-yang-json]. The fact that it can be used with different protocols and interfaces provides for a degree of "future-proofing" of model implementations. Also, YANG can serve as the basis for model-driven toolchains, such as used in the Open Daylight project.

The data model is defined in several YANG modules:

- Module "l3-unicast-igp-topology" defines a model for Layer 3 Unicast IGP topologies. To do so, it augments general network topology model defined in [I-D.draft-ietf-i2rs-yang-network-topo] with information specific to Layer 3 Unicast IGP. In doing so, it also illustrates the extension patterns associated with extending respectively augmenting the general topology model to meet the needs of a specific topology.

- Module "ospf-topology" defines a topology model for OSPF, building on and extending the Layer 3 Unicast IGP topology model. It serves as an example of how the general topology model can be refined across multiple levels.

- Module "isis-topology" defines a topology model for IS-IS, again building on and extending the Layer 3 Unicast IGP topology model.

Information that is kept in the Traffic Engineering Database (TED) is specified in a separate model and outside the scope of this specification.

2. Definitions and Acronyms

Datastore: A conceptual store of instantiated management information, with individual data items represented by data nodes which are arranged in hierarchical manner.

Data subtree: An instantiated data node and the data nodes that are hierarchically contained within it.

HTTP: Hyper-Text Transfer Protocol

IGP: Interior Gateway Protocol

IS-IS: Intermediate System to Intermediate System protocol

LSP: Label Switched Path

NETCONF: Network Configuration Protocol

OSPF: Open Shortest Path First, a link state routing protocol

URI: Uniform Resource Identifier

ReST: Representational State Transfer, a style of stateless interface and protocol that is generally carried over HTTP

SRLG: Shared Risk Link Group

TED: Traffic Engineering Database

YANG: A data definition language for NETCONF

3. Model overview

This section provides an overview of the Layer 3 network topology model.

3.1. Model structure

The network topology model is defined by the following YANG modules, whose relationship is roughly depicted in the figure below. The base network topology is included in the diagram for completeness.

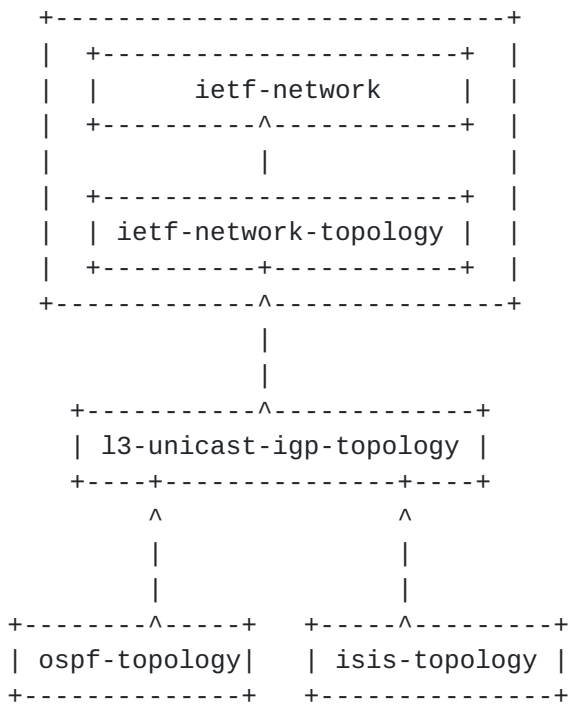

Figure 1: Overall model structure

YANG modules ietf-network and ietf-network-topology collectively define the basic network topology model. YANG module l3-unicast-igp-topology augments those models with additional definitions needed to represent Layer 3 Unicast IGP topologies. This module in turn is augmented by YANG modules with additional definitions for OSPF and for IS-IS topologies, ospf-topology and isis-topology, respectively.

3.2. Layer 3 Unicast - IGP

The Layer 3 Unicast IGP topology model is defined by YANG module "l3-unicast-igp-topology". The model is depicted in the following diagram. Brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. Notifications are not depicted. The prefix "nt:" refers to the YANG module for network topology.

```
module: l3-unicast-igp-topology
augment /nw:network/nw:network-types:
   +--rw l3-unicast-igp-topology!
augment /nw:network:
   +--rw igp-topology-attributes
      +--rw name?   string
      +--rw flag*   flag-type
augment /nw:network/nw:node:
   +--rw igp-node-attributes
      +--rw name?        inet:domain-name
      +--rw flag*        flag-type
      +--rw router-id*   inet:ip-address
      +--rw prefix* [prefix]
         +--rw prefix    inet:ip-prefix
         +--rw metric?   uint32
         +--rw flag*     flag-type
augment /nw:network/nt:link:
   +--rw igp-link-attributes
      +--rw name?     string
      +--rw flag*     flag-type
      +--rw metric?   uint32
augment /nw:network/nw:node/nt:termination-point:
   +--rw igp-termination-point-attributes
      +--rw (termination-point-type)?
         +--:(ip)
         |  +--rw ip-address*      inet:ip-address
         +--:(unnumbered)
            +--rw unnumbered-id?   uint32
```

The module augments the original ietf-network and ietf-network-topology modules as follows:

- A new network topology type is introduced, l3-unicast-igp-topology. The corresponding container augments the network-types of the ietf-network module.

- Additional topology attributes are introduced, defined in a grouping, which augments the "network" list of the network module.

The attributes include an IGP name, as well as a set of flags (represented through a leaf-list). Each type of flag is represented by a separate identity. This allows to introduce additional flags in augmenting modules that are associated with specific IGP topologies, without needing to revise this module.

- Additional data objects for nodes are introduced by augmenting the "node" list of the network module. New objects include again a set of flags, as well as a list of prefixes. Each prefix in turn includes an ip prefix, a metric, and a prefix-specific set of flags.

- Links (in the ietf-network-topology module) are augmented with a set of parameters as well, allowing to associate a link with an IGP name, another set of flags, and a link metric.

- Termination points (in the ietf-network-topology module as well) are augmented with a choice of IP address or identifier.

In addition, the module defines a set of notifications to alert clients of any events concerning links, nodes, prefixes, and termination points. Each notification includes an indication of the type of event, the topology from which it originated, and the affected node, or link, or prefix, or termination point. In addition, as a convenience to applications, additional data of the affected node, or link, or termination point (respectively) is included. While this makes notifications larger in volume than they would need to be, it avoids the need for subsequent retrieval of context information, which also might have changed in the meantime.

3.3. OSPF Topology

OSPF is the next type of topology represented in the model. OSPF represents a particular type of Layer 3 Unicast IGP. Accordingly, this time the Layer 3 Unicast IGP topology model needs to be extended. The corresponding extensions are introduced in a separate YANG module "ospf-topology", whose structure is depicted in the following diagram. For the most part, this module augments "l3-unicast-igp-topology". Like before, brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. Notifications respectively augmentations of notifications are not depicted.

```
   module: ospf-topology
   augment /nw:network/nw:network-types/l3t:l3-unicast-igp-topology:
      +--rw ospf!
   augment /nw:network/l3t:igp-topology-attributes:
      +--rw ospf-topology-attributes
         +--rw area-id?   area-id
   augment /nw:network/nw:node/l3t:igp-node-attributes:
      +--rw ospf-node-attributes
         +--rw (router-type)?
         |  +--:(abr)
         |  |  +--rw abr?                empty
         |  +--:(asbr)
         |  |  +--rw asbr?               empty
         |  +--:(internal)
         |  |  +--rw internal?           empty
         |  +--:(pseudonode)
         |     +--rw pseudonode?         empty
         +--rw dr-interface-id?     uint32
         +--rw multi-topology-id*   uint8
         +--rw capabilities?        bits
   augment /nw:network/nt:link/l3t:igp-link-attributes:
      +--rw ospf-link-attributes
         +--rw multi-topology-id?   uint8
   augment /nw:network/nw:node/l3t:igp-node-attributes/l3t:prefix:
      +--rw ospf-prefix-attributes
         +--rw forwarding-address?   inet:ipv4-address
```

The module augments "l3-unicast-igp-topology" as follows:

- A new topology type for an OSPF topology is introduced.

- Additional topology attributes are defined in a new grouping which augments igp-topology-attributes of the l3-unicast-igp-topology module. The attributes include an OSPF area-id identifying the OSPF area.

- Additional data objects for nodes are introduced by augmenting the igp-node-attributes of the l3-unicast-igp-topology module. New objects include router-type, dr-interface-id for pseudonodes, list of multi-topology-ids, ospf node capabilities, and traffic engineering attributes.

- Links are augmented with a multi-topology-id and traffic engineering link attributes.

- Prefixes are augmented with OSPF specific forwarding address.

In addition, the module extends IGP node, link and prefix notifications with OSPF attributes.

3.4. IS-IS Topology

IS-IS is another type of Layer 3 Unicast IGP. Like OSPF topology, IS-IS topology is defined in a separate module, "isis-topology", which augments "l3-unicast-igp-topology". The structure is depicted in the following diagram. Like before, brackets enclose list keys, "rw" means configuration, "ro" operational state data, "?" designates optional nodes, "*" designates nodes that can have multiple instances. Parantheses enclose choice and case nodes. Notifications are not depicted.

```
module: isis-topology
augment /nw:network/nw:network-types/l3t:l3-unicast-igp-topology:
   +--rw isis!
augment /nw:network/l3t:igp-topology-attributes:
   +--rw isis-topology-attributes
      +--rw net?   iso-net-id
augment /nw:network/nw:node/l3t:igp-node-attributes:
   +--rw isis-node-attributes
      +--rw iso
      |  +--rw iso-system-id?       iso-system-id
      |  +--rw iso-pseudonode-id?   iso-pseudonode-id
      +--rw net*                 iso-net-id
      +--rw multi-topology-id*   uint8
      +--rw (router-type)?
         +--:(level-2)
         |  +--rw level-2?            empty
         +--:(level-1)
         |  +--rw level-1?            empty
         +--:(level-1-2)
            +--rw level-1-2?           empty
augment /nw:network/nt:link/l3t:igp-link-attributes:
   +--rw isis-link-attributes
      +--rw multi-topology-id?   uint8
```

The module augments the l3-unicast-igp-topology as follows:

- A new topology type is introduced for isis.

- Additional topology attributes are introduced in a new grouping which augments "igp-topology-attributes" of the l3-unicast-igp-topology module. The attributes include an ISIS NET-id identifying the area.

- Additional data objects for nodes are introduced by augmenting "igp-node-attributes" of the l3-unicast-igp-topology module. New objects include router-type, iso-system-id to identify the router, a list of multi-topology-id, a list of NET ids, and traffic engineering attributes.

- Links are augmented with multi-topology-id and traffic engineering link attributes.

In addition, the module augments IGP nodes and links with ISIS attributes.

4. Layer 3 Unicast IGP Topology YANG Module

```
<CODE BEGINS>
file "l3-unicast-igp-topology@2015-06-08.yang"
module l3-unicast-igp-topology {
  yang-version 1;
  namespace "urn:ietf:params:xml:ns:yang:l3-unicast-igp-topology";
  prefix "l3t";
  import ietf-network {
    prefix "nw";
  }
  import ietf-network-topology {
    prefix "nt";
  }
  import ietf-inet-types {
    prefix "inet";
  }

  organization "TBD";
  contact "TBD";

  description
    "This module defines a model for the layer 3 IGP topology.";

  revision "2015-06-08" {
    description "Initial revision";
    reference "TBD";
  }

  typedef igp-event-type {
    type enumeration {
      enum "add" {
        value 0;
        description
          "An IGP node or link or prefix or termination-point has
          been added";
```

```
        }
        enum "remove" {
          value 1;
          description
            "An IGP node or link or prefix or termination-point has
            been removed";
        }
        enum "update" {
          value 2;
          description
            "An IGP node or link or prefix or termination-point has
            been updated";
        }
      }
      description "IGP Event type for notifications";
    } // igp-event-type

    identity flag-identity {
      description "Base type for flags";
    }
    identity undefined-flag {
      base "flag-identity";
      description "Undefined flag";
    }

    typedef flag-type {
      type identityref {
        base "flag-identity";
      }
      description "Type for flags";
    }

    grouping network-ref {
      description
        "Grouping for an absolute reference to a network topology
        instance.";
      leaf network-ref {
        type leafref {
          path "/nw:network/nw:network-id";
        }
        description
          "An absolute reference to a network topology instance.";
      }
    }

    grouping link-ref {
      description
        "Grouping for an absolute reference to a link instance.";
```

```
        uses network-ref;
        leaf link-ref {
          type leafref {
            path "/nw:network"
              +"[nw:network-id = current()/../network-ref]"
              +"/nt:link/nt:link-id";
          }
        description
          "An absolute reference to a link instance.";
      }
    }

    grouping node-ref {
      description
        "Grouping for an absolute reference to a node instance.";
      uses network-ref;
      leaf node-ref {
        type leafref {
          path "/nw:network"
            +"[nw:network-id = current()/../network-ref]"
            +"/nw:node/nw:node-id";
        }
        description
          "An absolute reference to a node instance.";
      }
    }

    grouping tp-ref {
      description
        "Grouping for an absolute reference to a termination point.";
      uses node-ref;
      leaf tp-ref {
        type leafref {
          path "/nw:network"
            +"[nw:network-id = current()/../network-ref]"
            +"/nw:node[nw:node-id = current()/../node-ref]"
            +"/nt:termination-point/nt:tp-id";
        }
        description
          "Grouping for an absolute reference to a termination point.";
      }
    }

    grouping igp-prefix-attributes {
      description
        "IGP prefix attributes";
      leaf prefix {
        type inet:ip-prefix;
```

```
        description "IP prefix value";
      }
      leaf metric {
        type uint32;
        description "Prefix metric";
      }
      leaf-list flag {
        type flag-type;
        description "Prefix flags";
      }
    }

    grouping l3-unicast-igp-topology-type {
      description "Identify the topology type to be L3 unicast.";
      container l3-unicast-igp-topology {
        presence "indicates L3 Unicast IGP Topology";
        description
          "The presence of the container node indicates L3 Unicast
          IGP Topology";
      }
    }

    grouping igp-topology-attributes {
      description "Topology scope attributes";
      container igp-topology-attributes {
        description "Containing topology attributes";
        leaf name {
          type string;
          description "Name of the topology";
        }
        leaf-list flag {
          type flag-type;
          description "Topology flags";
        }
      }
    }

    grouping igp-node-attributes {
      description "IGP node scope attributes";
      container igp-node-attributes {
        description "Containing node attributes";
        leaf name {
          type inet:domain-name;
          description "Node name";
        }
        leaf-list flag {
          type flag-type;
          description "Node operational flags";
```

```
        }
        leaf-list router-id {
          type inet:ip-address;
          description "Router-id for the node";
        }
        list prefix {
          key "prefix";
          description
            "A list of prefixes along with their attributes";
          uses igp-prefix-attributes;
        }
      }
    }

    grouping igp-link-attributes {
      description "IGP link scope attributes";
      container igp-link-attributes {
        description "Containing link attributes";
        leaf name {
          type string;
          description "Link Name";
        }
        leaf-list flag {
          type flag-type;
          description "Link flags";
        }
        leaf metric {
          type uint32 {
            range "0..16777215"  {
              description "
                          ";
              // OSPF/ISIS supports max 3 byte metric.
              // Ideally we would like this restriction to be
              // defined in the derived models, however,
              // we are not allowed to augment a "must" statement.
            }
          }
          description "Link Metric";
        }
      }
    } // grouping igp-link-attributes

    grouping igp-termination-point-attributes {
      description "IGP termination point scope attributes";
      container igp-termination-point-attributes {
        description "Containing termination point attributes";
        choice termination-point-type {
          description "Indicates the termination point type";
```

```
          case ip {
            leaf-list ip-address {
              type inet:ip-address;
              description "IPv4 or IPv6 address";
            }
          }
          case unnumbered {
            leaf unnumbered-id {
              type uint32;
              description "Unnumbered interface identifier";
            }
          }
        }
      }
    } // grouping igp-termination-point-attributes

    augment "/nw:network/nw:network-types" {
      description
          "Introduce new network type for L3 unicast IGP topology";
      uses l3-unicast-igp-topology-type;
    }

    augment "/nw:network" {
      when "nw:network-types/l3-unicast-igp-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast IGP topology";
      }
      description
          "Configuration parameters for L3 unicast IPG for the network
          as a whole";
      uses igp-topology-attributes;
    }

    augment "/nw:network/nw:node" {
      when "../nw:network-types/l3-unicast-igp-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast IGP topology";
      }
      description
          "Configuration parameters for L3 unicast IPG at the node
          level";
      uses igp-node-attributes;
    }

    augment "/nw:network/nt:link" {
      when "../nw:network-types/l3-unicast-igp-topology" {
```

```
        description
          "Augmentation parameters apply only for networks with
          L3 unicast IGP topology";
      }
      description "Augment topology link configuration";
      uses igp-link-attributes;
    }

    augment "/nw:network/nw:node/"
          +"nt:termination-point" {
      when "../../nw:network-types/l3-unicast-igp-topology" {
        description
          "Augmentation parameters apply only for networks with
          L3 unicast IGP topology";
      }
      description "Augment topology termination point configuration";
      uses igp-termination-point-attributes;
    }

    notification igp-node-event {
      description "Notification event for IGP node";
      leaf igp-event-type {
        type igp-event-type;
        description "Event type";
      }
      uses node-ref;
      uses l3-unicast-igp-topology-type;
      uses igp-node-attributes;
    }

    notification igp-link-event {
      description "Notification event for IGP link";
      leaf igp-event-type {
        type igp-event-type;
        description "Event type";
      }
      uses link-ref;
      uses l3-unicast-igp-topology-type;
      uses igp-link-attributes;
    }

    notification igp-prefix-event {
      description "Notification event for IGP prefix";
      leaf igp-event-type {
        type igp-event-type;
        description "Event type";
      }
      uses node-ref;
```

```
      uses l3-unicast-igp-topology-type;
      container prefix {
        description "Containing IPG prefix attributes";
        uses igp-prefix-attributes;
      }
    }

    notification termination-point-event {
      description "Notification event for IGP termination point";
      leaf igp-event-type {
        type igp-event-type;
        description "Event type";
      }
      uses tp-ref;
      uses l3-unicast-igp-topology-type;
      uses igp-termination-point-attributes;
    }
  }

  <CODE ENDS>
```

5. OSPF Topology YANG Module

```
  <CODE BEGINS>
  file "ospf-topology@2015-06-08.yang"
  module ospf-topology {
    yang-version 1;
    namespace "urn:ietf:params:xml:ns:yang:ospf-topology";
    prefix "ospf";

    import ietf-inet-types {
      prefix "inet";
    }
    import ietf-network {
      prefix "nw";
    }
    import ietf-network-topology {
      prefix "nt";
    }
    import l3-unicast-igp-topology {
      prefix "l3t";
    }

    organization "TBD";
    contact "TBD";
    description "OSPF Topology model";

    revision "2015-06-08" {
```

```
      description "Initial revision";
      reference "TBD";
    }

    typedef area-id {
      type uint32;
      description "OSPF Area ID";
    }

    grouping ospf-topology-type {
      description
        "Identifies the OSPF topology type.";
      container ospf {
        presence "indiates OSPF Topology";
        description
          "Its presence identifies the OSPF topology type.";
      }
    }

    augment "/nw:network/nw:network-types/"
      + "l3t:l3-unicast-igp-topology" {
      description
        "Defines the OSPF topology type.";
      uses ospf-topology-type;
    }

    augment "/nw:network/l3t:igp-topology-attributes" {
      when "../nw:network-types/l3t:l3-unicast-igp-topology/ospf" {
        description "Augment only for OSPF topology";
      }
      description "Augment topology configuration";
      container ospf-topology-attributes {
        description "Containing topology attributes";
        leaf area-id {
          type area-id;
          description "OSPF area ID";
        }
      }
    }

    augment "/nw:network/nw:node/l3t:igp-node-attributes" {
      when "../../nw:network-types/l3t:l3-unicast-igp-topology/ospf" {
        description "Augment only for OSPF topology";
      }
      description "Augment node configuration";
      uses ospf-node-attributes;
    }
```

```
    augment "/nw:network/nt:link/l3t:igp-link-attributes" {
      when "../../nw:network-types/l3t:l3-unicast-igp-topology/ospf" {
        description "Augment only for OSPF topology";
      }
      description "Augment link configuration";
      uses ospf-link-attributes;
    }

    augment "/nw:network/nw:node/l3t:igp-node-attributes/l3t:prefix" {
      when "../../../nw:network-types/l3t:l3-unicast-igp-topology/"
        +"ospf" {
        description "Augment only for OSPF topology";
      }
      description "Augment prefix";
      uses ospf-prefix-attributes;
    }

    grouping ospf-node-attributes {
      description "OSPF node scope attributes";
      container ospf-node-attributes {
        description "Containing node attributes";
        choice router-type {
          description "Indicates router type";
          case abr {
            leaf abr {
              type empty;
              description "The node is ABR";
            }
          }
          case asbr {
            leaf asbr {
              type empty;
              description "The node is ASBR";
            }
          }
          case internal {
            leaf internal {
              type empty;
              description "The node is internal";
            }
          }
          case pseudonode {
            leaf pseudonode {
              type empty;
              description "The node is pseudonode";
            }
          }
        }
```

```
        leaf dr-interface-id {
          when "../router-type/pseudonode" {
            description "Valid only for pseudonode";
          }
          type uint32;
          default "0";
          description "For pseudonodes, DR interface-id";
        }
        leaf-list multi-topology-id {
          type uint8 {
            range "0..127";
          }
          max-elements "128";
          description
            "List of Multi-Topology Identifier up-to 128 (0-127).
            RFC 4915";
        }
        leaf capabilities {
          type bits {
            bit graceful-restart-capable {
              position 0;
              description "Graceful restart capable";
            }
            bit graceful-restart-helper {
              position 1;
              description "Graceful restart helper";
            }
            bit stub-router-support {
              position 2;
              description "Stub router support";
            }
            bit traffic-engineering-support {
              position 3;
              description "Traffic engineering support";
            }
            bit point-to-point-over-lan {
              position 4;
              description "Support point to point over LAN";
            }
            bit experimental-te {
              position 5;
              description "Support experimental traffic engineering";
            }
          }
          description "OSPF capabilities as bit vector. RFC 4970";
        }
      } // ospf
    } // ospf-node-attributes
```

```
    grouping ospf-link-attributes {
      description "OSPF link scope attributes";
      container ospf-link-attributes {
        description "Containing OSPF link attributes";
        leaf multi-topology-id {
          type uint8 {
            range "0..127";
          }
          description "Muti topology ID";
        }
      }
    } // ospf-link-attributes

    grouping ospf-prefix-attributes {
      description "OSPF prefix attributes";
      container ospf-prefix-attributes {
        description "Containing prefix attributes";
        leaf forwarding-address {
          when "../../l3t:l3-unicast-igp-topology/l3t:ospf/"
            +"l3t:router-type/l3t:asbr" {
            description "Valid only for ABSR";
          }
          type inet:ipv4-address;
          description "Forwarding address for ABSR";
        }
      }
    }

    augment "/l3t:igp-node-event" {
      description "OSPF node event";
      uses ospf-topology-type;
      uses ospf:ospf-node-attributes;
    }

    augment "/l3t:igp-link-event" {
      description "OSPF link event";
      uses ospf-topology-type;
      uses ospf:ospf-link-attributes;
    }

    augment "/l3t:igp-prefix-event" {
      description "OSPF prefix event";
      uses ospf-topology-type;
      uses ospf:ospf-prefix-attributes;
    }
  }
  <CODE ENDS>
```

6. ISIS Topology YANG Module

```
<CODE BEGINS>
file "isis-topology@2015-06-08.yang"
module isis-topology {
  yang-version 1;
  namespace "urn:ietf:params:xml:ns:yang:isis-topology";
  prefix "isis";

  import ietf-network {
    prefix "nw";
  }
  import ietf-network-topology {
    prefix "nt";
  }
  import l3-unicast-igp-topology {
    prefix "l3t";
  }

  organization "TBD";
  contact "TBD";
  description "ISIS Topology model";

  revision "2015-06-08" {
    description "Initial version";
    reference "TBD";
  }
  typedef iso-system-id {
    type string {
      pattern '[0-9a-fA-F]{4}(\.[0-9a-fA-F]{4}){2}';
    }
    description "ISO System ID. RFC 1237";
  }

  typedef iso-pseudonode-id {
    type string {
      pattern '[0-9a-fA-F]{2}';
    }
    description "ISO pseudonode id for broadcast network";
  }

  typedef iso-net-id {
    type string {
      pattern '[0-9a-fA-F]{2}((\.[0-9a-fA-F]{4}){6})';
    }
    description "ISO NET ID. RFC 1237";
  }
```

```
    grouping isis-topology-type {
      description
        "Identifies the ISIS topology type.";
      container isis {
        presence "Indicates ISIS Topology";
        description
          "Its presence identifies the ISIS topology type.";
      }
    }

    augment "/nw:network/nw:network-types/"
      +"l3t:l3-unicast-igp-topology" {
      description
        "Defines the ISIS topology type.";
      uses isis-topology-type;
    }

    augment "/nw:network/l3t:igp-topology-attributes" {
      when "../nw:network-types/l3t:l3-unicast-igp-topology/isis" {
        description "Augment only for ISIS topology";
      }
      description "Augment topology configuration";
      container isis-topology-attributes {
        description "Containing topology attributes";
        leaf net {
          type iso-net-id;
          description "ISO NET ID value";
        }
      }
    }

    augment "/nw:network/nw:node/"
      +"l3t:igp-node-attributes" {
      when "../../nw:network-types/l3t:l3-unicast-igp-topology/isis" {
        description "Augment only for ISIS topology";
      }
      description "Augment node configuration";
      uses isis-node-attributes;
    }

    augment "/nw:network/nt:link/l3t:igp-link-attributes" {
      when "../../nw:network-types/l3t:l3-unicast-igp-topology/isis" {
        description "Augment only for ISIS topology";
      }
      description "Augment link configuration";
      uses isis-link-attributes;
    }
```

```
     grouping isis-node-attributes {
       description "ISIS node scope attributes";
       container isis-node-attributes {
         description "Containing node attributes";
         container iso {
           description "Containing ISO atrributes";
           leaf iso-system-id {
             type iso-system-id;
             description "ISO system ID";
           }
           leaf iso-pseudonode-id {
             type iso-pseudonode-id;
             default "00";
             description "Pseudonode ID";
           }
         }
         leaf-list net {
           type iso-net-id;
           max-elements 3;
           description "List of ISO NET IDs";
         }
         leaf-list multi-topology-id {
           type uint8 {
             range "0..127";
           }
           max-elements "128";
           description
             "List of Multi Topology Identifier upto 128 (0-127).
             RFC 4915";
         }
         choice router-type {
           description "Indicates router type";
           case level-2 {
             leaf level-2 {
               type empty;
               description "Level-2 only";
             }
           }
           case level-1 {
             leaf level-1 {
               type empty;
               description "Level-1 only";
             }
           }
           case level-1-2 {
             leaf level-1-2 {
               type empty;
               description "Level-1 and Level-2";
```

```
            }
          }
        }
      }
    }

    grouping isis-link-attributes {
      description "ISIS link scope attributes";
      container isis-link-attributes {
        description "Containing link attributes";
        leaf multi-topology-id {
          type uint8 {
            range "0..127";
          }
          description "Muti topology ID";
        }
      }
    }

    augment "/l3t:igp-node-event" {
      description "ISIS node event";
      uses isis-topology-type;
      uses isis-node-attributes;
    }

    augment "/l3t:igp-link-event" {
      description "ISIS link event";
      uses isis-topology-type;
      uses isis-link-attributes;
    }
  } // Module isis-topology

  <CODE ENDS>
```

7. Security Considerations

The transport protocol used for sending the topology data MUST support authentication and SHOULD support encryption. The data-model by itself does not create any security implications.

8. Contributors

The model presented in this paper was contributed to by more people than can be listed on the author list. Additional contributors include:

- Ken Gray, Juniper Networks

- Tom Nadeau, Brocade

- Aleksandr Zhdankin, Cisco

9. Acknowledgements

We wish to acknowledge the helpful contributions, comments, and suggestions that were received from Ladislav Lhotka, Andy Bierman, Carlos Pignataro, Joel Halpern, Juergen Schoenwaelder, Alia Atlas, and Susan Hares.

10. References

10.1. Normative References

[I-D.draft-ietf-i2rs-yang-network-topo]
Clemm, A., Medved, J., Tkacik, T., Varga, R., Bahadur, N., and H. Ananthakrishnan, "A YANG Data Model for Network Topologies", I-D draft-ietf-i2rs-yang-network-topo-01, June 2015.

[I-D.draft-ietf-netconf-restconf]
Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", I-D draft-ietf-netconf-restconf-05, June 2015.

[I-D.draft-ietf-netmod-yang-json]
Lhotka, L., "JSON Encoding of Data Modeled with YANG", I-D draft-ietf-netmod-yang-json-03, February 2015.

[RFC1195] Callon, R., "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments", RFC 1195, December 1990.

[RFC2178] Moy, J., "OSPF Version 2", RFC 2178, July 1997.

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

[RFC6021] Schoenwaelder, J., "Common YANG Data Types", RFC 6021, October 2010.

[RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A. Bierman, "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

[RFC7223] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 7223, May 2014.

10.2. Informative References

[I-D.draft-ietf-i2rs-usecase-reqs-summary]
Hares, S. and M. Chen, "Summary of I2RS Use Case Requirements", I-D draft-ietf-i2rs-usecase-reqs-summary-01, May 2015.

Authors' Addresses

Alexander Clemm
Cisco

EMail: alex@cisco.com

Jan Medved
Cisco

EMail: jmedved@cisco.com

Robert Varga
Pantheon Technologies SRO

EMail: robert.varga@pantheon.sk

Tony Tkacik
Pantheon Technologies SRO

EMail: tony.tkacik@pantheon.sk

Xufeng Liu
Ericsson

EMail: xufeng.liu@ericsson.com

Igor Bryskin
Adva Optical

EMail: ibryskin@advaoptical.com

Aihua Guo
Adva Optical

EMail: aguo@advaoptical.com

Hariharan Ananthakrishnan
Packet Design

EMail: hari@packetdesign.com

Nitin Bahadur
Bracket Computing

EMail: nitin_bahadur@yahoo.com

Vishnu Beeram
Juniper Networks

EMail: vbeeram@juniper.net