                          YANG Module Tags
                 draft-rtgyangdt-netmod-module-tags-00

Abstract

   This document defines two modules that support the association of
   tags with modules. Tags may be included in a module or associated
   with a module through the use of an augmentation to YANG library that
   is defined in this document. The expectation is for such tags to be
   used to help classify and organize modules. Tags may be standardized
   and assigned during module definition; assigned by implementations;
   or dynamically defined and set by users. This document provides
   guidance to future model writers and, as such, this document updates
   [I-D.ietf-netmod-rfc6087bis].

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 August 12, 2017.

publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   The use of tags for classification and organization is fairly
   ubiquitous not only within IETF protocols, but in the internet itself
   (see #hashtags). Tags can be usefully standardized, but they can
   also serve as a non-standardized mechanism available for users to
   define themselves. Our solution provides for both cases allowing for

the most flexibility. In particular, tags may be standardized and
assigned during module definition; assigned by implementations; or
dynamically defined and set by users.

This document defines two modules that support the association of
tags with modules. The first module defines a grouping which
contains a list of tags as well as rpc statements for changing the
contents of the list. Tags are strings that are structured to enable
the differentiation of globally assigned and non-assigned tags based
on a fixed prefix. This document also defines an initial set of
globally assigned tags.

The second module defined in this document defines an augmentation to
YANG Library [RFC7895]. It uses (imports) the first module to
provide a well known location for tags.

Section 9 provides guidelines for authors of YANG data models. This
section updates [I-D.ietf-netmod-rfc6087bis].

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

Note that lower case versions of these key words are used in section
Section 9 where guidance is provided to future document authors.

3. Tag Locations

Two tag list locations are defined. One location is within the
module itself, and the other location is in the yang library under
the modules entry. When a module includes tags, the same tag list
may also be presented in yang library.

To add tags to a module, the module definition includes a tag list
using the 'module-tags' grouping defined in this document. This list
MUST be added by a module author under container named "module-tags"
at the root of their module.

4. Tag Prefixes

All tags have a prefix indicating who owns their definition. An IANA
registry is used to support standardizing tag prefixes. Currently 2
prefixes are defined with all others reserved.

4.1. IETF Standard Tags

   An IETF standard tag is a tag that has the prefix "ietf:". All IETF
   standard tags are registered with IANA in a registry defined later in
   this document.

4.2. Vendor Tags

   A vendor tag is a tag that has the prefix "vendor:". These tags are
   defined by the vendor that implements the module, and are not
   standardized.

4.3. Local Tags

   A local tag is any tag that has the prefix "local:". These tags are
   defined by the local user/administrator, and will never be
   standardized.

4.4. Reserved Tags

   Any tag not starting with the prefix "ietf:", "vendor:" or "local:"
   is reserved for future standardization.

5. Tag Management

   Tags can become associated with a module in a number of ways. Tags
   may be defined as associated at model design time, at implementation
   time, or via user administrative control. As the main consumer of
   tags are users, users may remove any tag, no matter how the tag
   became associated with a module.

5.1. Module Definition Association

   A module definition SHOULD indicate a set of standard tags to be
   automatically added by the module implementer. These tags MUST be
   standard tags (Section 4.1). This does imply that new modules may
   also drive the addition of new standard tags to the IANA registry.

5.2. Implementation Association

   An implementation MAY include additional tags associated with a
   module. These tags may be standard or vendor specific tags.

5.3. Administrative Tagging

   RPC statements are defined in this document to enable administrative
   addition and removal of tags from a module by a user. An additional

rpc is defined to reset a module's tag list to the implementation
default.

Each rpc identifies the module on which the tag operation is to be
performed. This identifications reuses the format of the common-
leafs (sub) grouping defined in [RFC7895]. The grouping itself is
refined in Section 6 so that it is a stand-alone grouping.

Implementations that support the rpc statements defined in this
document MUST ensure that a specific module's tags leaf list is
consistent across any location from which the list is available.
Specifically this includes in the module itself, per Section 9.1, or
in yang library, per Section 7.

Implementations that do not support the defined rpc statements
(whether at all, or just for a particular rpc or module) MUST respond
with an YANG transport protocol-appropriate rpc layer error when such
a statement is received.

5.3.1. Adding Tags

The "add-tags" rpc statement is defined to support the addition of
tags. This rpc statement takes as input module identification
information and the list of tags to add.

No restriction is placed on the tag values to add.

5.3.2. Removing Tags

The "remove-tags" rpc statement is defined to remove tags. This rpc
statement takes as input module identification information and the
list of tags to remove.

No restriction is placed on the tag values to remove. This means
that tags associated based on a module's definition or implementation
MUST be removable.

5.3.3. Resetting Tags

The "reset-tags" rpc statement is defined to reset a module's tag
list to the implementation default, i.e. the tags that are present
based on module definition and any that are added during
implementation time. This rpc statement takes module identification
information as input, and provides the list of list of tags that are
present after the reset.

6. Tags Module Structure

6.1. Tags Module Tree

   The tree associated with the tags module is:

```
module: ietf-module-tags
  rpcs:
    +---x add-tags
    |  +---w input
    |     +---w name       yang:yang-identifier
    |     +---w revision?  union
    |     +---w tags*       string
    +---x remove-tags
    |  +---w input
    |     +---w name       yang:yang-identifier
    |     +---w revision?  union
    |     +---w tags*       string
    +---x reset-tags
       +---w input
       |  +---w name       yang:yang-identifier
       |  +---w revision?  union
       +--ro output
          +--ro tags*    string
```

6.2. Tags Module

```
<CODE BEGINS> file "ietf-module-tags@2017-02-08.yang"
module ietf-module-tags {
  yang-version "1.1";
  namespace "urn:ietf:params:xml:ns:yang:ietf-module-tags";
  prefix "mtags";

  import ietf-yang-types {
    prefix yang;
  }

  import ietf-yang-library {
    prefix yanglib;
  }

  // meta
  organization "IETF NetMod Working Group (NetMod)";

  contact
    "NetMod Working Group - <netmod@ietf.org>";

  description
```

```
      "This module describes a tagging mechanism for yang module.
       Tags may be IANA assigned or privately defined types.";

    revision "2017-02-08" {
      description
        "Initial revision.";
      reference "TBD";
    }

    grouping module-tags {
      description
        "A grouping that may be used to classify a module.";

      leaf-list tags {
        type string;

        config false;

        description
          "The module associated tags. See the IANA 'YANG Module Tag
           Prefix' registry for reserved prefixes and the IANA 'YANG
           Module IETF Tag' registry for IETF standard tags";
      }
    }

    grouping yanglib-common-leafs {
      description
        "Common parameters for YANG modules and submodules.
         This definition extract from RFC7895 as it is defined as
         a grouping within a grouping.

         TBD is there a legal way to use a grouping defined wuthin
         another grouping without using the parent? If so, should change
         to that.";

      leaf name {
        type yang:yang-identifier;
        mandatory true;
        description
          "The YANG module or submodule name.";
      }
      leaf revision {
        type union {
          type yanglib:revision-identifier;
          type string { length 0; }
        }
        description
          "The YANG module or submodule revision date.
```

```
            A zero-length string is used if no revision statement
            is present in the YANG module or submodule.";
      }
    }

    rpc add-tags {
      description
        "Add a list of tags to a given module.";

      input {
        uses yanglib-common-leafs;
        uses module-tags;
      }
    }

    rpc remove-tags {
      description
        "Remove a list of tags, if present, from a given module.";

      input {
        uses yanglib-common-leafs;
        uses module-tags;
      }
    }

    rpc reset-tags {
      description
        "Reset a list of tags for a given module to the list of module
         and implementation time defiend tags. It provides the list of
         tags associated with the module post reset.";

      input {
        uses yanglib-common-leafs;
      }

      output {
          uses module-tags;
      }
    }
  }
  <CODE ENDS>
```

7. Library Augmentation

    Tags can also be associated with a module using the yang library
    [RFC7895]. When a server supports both yang library and the
    augmentation defined below, a user can add, remove and search for
    tags for any module on the server regardless of whether the specific

module included tag support in its definition or not. If a server
supports ietf-module-tags and the yang library it SHOULD also support
the ietf-library-tags module.

The tree associated with the defined augmentation is:

```
module: ietf-library-tags
  augment /yanglib:modules-state/yanglib:module:
    +--ro tags*    string
```

7.1. Library Augmentation Module

```
   <CODE BEGINS> file "ietf-library-tags@2017-02-08.yang"
   module ietf-library-tags {
     // namespace
     namespace "urn:ietf:params:xml:ns:yang:ietf-library-tags";

     prefix ylibtags;

     import ietf-yang-library {
        prefix yanglib;
     }
     import ietf-module-tags {
        prefix mtags;
     }

     // meta
     organization "IETF NetMod Working Group (NetMod)";

     contact
         "NetMod Working Group - <netmod@ietf.org>";

     description
       "This module augments ietf-yang-library with searchable
       classfication tags.  Tags may be IANA or privately defined
       types.";

     revision "2017-02-06" {
       description
         "Initial revision.";
       reference "RFC TBD";
     }

     augment "/yanglib:modules-state/yanglib:module" {
       description
         "The yang library structure is augmented with a module tags
          list. This allows operators to tag modules regardless of
          whether the modules included tag support or not";

       uses mtags:module-tags;

     }
   }
   <CODE ENDS>
```

8. Other Classifications

   It's worth noting that a different yang module classification
   document exists [I-D.ietf-netmod-yang-model-classification]. That
   document is classifying modules in only a logical manner and does not

define tagging or any other mechanisms. It divides yang modules into
2 categories (service or entity) and then into one of 3 origins:
standard, vendor or user. It does provide a good way to discuss and
identify modules in general. This document defines standard tags to
support [I-D.ietf-netmod-yang-model-classification] style
classification.

9. Guidelines to Model Writers

   This section updates [I-D.ietf-netmod-rfc6087bis]. This document
   makes two recommendations to model writers,

9.1. Include Module Tags

   The correct way to use the module-tags grouping is to include it in a
   standard location at the top level of your module, specifically
   contained within a container named "module-tags". This standard
   location allows searching module using a well-known xpath wilcard
   path. For example:

```
module sample-module {
  ...
  import ietf-module-tags {
    prefix mtags;
  }
  ...
  container module-tags {
    description
      "A list of classification tags associated with this
       module. The following predefined tags <MUST|SHOULD|MAY>
       be included by an implementation:
         - ietf:foo
         - ietf:bar
         - ...
      ";
    uses mtags:module-tags;
  }
  ...
}
```

   The associated tree will look like:

```
        module: sample-module
            +--rw module-tags
            |  +--ro tags*    string
            +--...
```

9.2. Define Standard Tags

A module should indicate, in the description of the "module-tags"
container, the set of tags that are to be populated in the leaf-list
for any implementation of the module. This description should also
include the appropriate conformance statement or statements, using
[RFC2119] language, for each tag.

The module writer may use existing standard tags, or use new tags
defined in the model definition, as appropriate. New tags should be
assigned in the IANA registry defined below, see Section 10.2 below.

10. IANA Considerations

10.1. YANG Module Tag Prefix Registry

This registry allocates tag prefixes. All YANG module tags must
begin with one of the prefixes in this registry.

The allocation policy for this registry is Specification Required
[RFC5226].

The initial values for this registry are as follows.

```
   prefix     description
   --------   ----------------------------------------------------
   ietf:      IETF Standard Tag allocated in the IANA YANG Module
              IETF Tag Registry.
   vendor:    Non-standardized tags allocated by the module implementer.
   local:     Non-standardized tags allocated by and for the user.
```

10.2. YANG Module IETF Tag Registry

This registry allocates prefixes that have the standard prefix
"ietf:". New values should be well considered and not achievable
through a combination of already existing standard tags.

The allocation policy for this registry is IETF Review [RFC5226].

The initial values for this registry are as follows.

[Editor's note: some of these tags are expected to move to
[I-D.ietf-rtgwg-device-model] if/when this document becomes a WG
document and that document is refactored to use tags.]

Tag	Description	Reference
ietf:area:art	Applications and Real-Time Area module.	[This document]
ietf:area:gen	General Area module.	[This document]
ietf:area:int	Internet Area module.	[This document]
ietf:area:ops	Operations and Management Area module.	[This document]
ietf:area:rtg	Routing Area module.	[This document]
ietf:area:sec	Security Area module.	[This document]
ietf:area:tsv	Transport Area module.	[This document]
ietf:entity	A module for an entity (*).	[This document]
ietf:service	A module for a service (*).	[This document]
ietf:hardware	A module for hardware.	[This document]
ietf:software	A module for software.	[This document]
ietf:protocol	A module representing a protocol.	[This document]
ietf:protocol:system-management	A module representing a	[This document]

	system management protocol.		
ietf:protocol:network-service	A module representing a network service protocol.	[This document]	
ietf:protocol:routing	A module representing a control plane routing protocol.	[This document]	
ietf:protocol:signaling	A module representing a control plane signaling protocol.	[This document]	
ietf:protocol:oam	A module representing a Operations, Administration, and Maintenance protocol.	[This document]	
ietf:protocol:lmp	A module representing a link management protocol.	[This document]	
ietf:protocol:routing:igp	An IGP protocol module.	[This document]	
ietf:protocol:routing:egp	An EGP protocol module.	[This document]	

(*) - see [I-D.ietf-netmod-yang-model-classification]

Table 1: IETF Module Tag Registry

11. References

11.1. Normative References

   [I-D.ietf-netmod-rfc6087bis]
              Bierman, A., "Guidelines for Authors and Reviewers of YANG
              Data Model Documents", draft-ietf-netmod-rfc6087bis-10
              (work in progress), January 2017.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an
              IANA Considerations Section in RFCs", BCP 26, RFC 5226,
              DOI 10.17487/RFC5226, May 2008,
              <http://www.rfc-editor.org/info/rfc5226>.

   [RFC7895] Bierman, A., Bjorklund, M., and K. Watsen, "YANG Module
              Library", RFC 7895, DOI 10.17487/RFC7895, June 2016,
              <http://www.rfc-editor.org/info/rfc7895>.

11.2. Informative References

   [I-D.ietf-netmod-yang-model-classification]
              Bogdanovic, D., Claise, B., and C. Moberg, "YANG Module
              Classification", draft-ietf-netmod-yang-model-
              classification-04 (work in progress), October 2016.

   [I-D.ietf-rtgwg-device-model]
              Lindem, A., Berger, L., Bogdanovic, D., and C. Hopps,
              "Network Device YANG Organizational Models", draft-ietf-
              rtgwg-device-model-01 (work in progress), October 2016.

Authors' Addresses

   Christan Hopps
   Deutsche Telekom

   Email: chopps@chopps.org

   Lou Berger
   LabN Consulting, L.L.C.

   Email: lberger@labn.net

   Dean Bogdanovic

   Email: ivandean@gmail.com