
YANG Schema Mount

draft-ietf-schema-mount-02

Martin Björklund

<mbj@tail-f.com>

Ladislav Lhotka

<lhotka@nic.cz>

18 July 2016

Objectives

Provide a data modelling mechanism for defining compound schemas:
a schema is embedded at a specific location of another schema.

```
+--rw if:interfaces
|   +--rw if:interface* [name]
|   |   ...
|   |   +--rw ip:ipv4
|   |   |   ...
|   |   +--rw ip:ipv6
|   |   ...
|   + ...
+--rw logical-device* [name]
+--rw name
|   ...
+--rw if:interfaces
+--rw if:interface* [name]
|   ...
|   +--rw ip:ipv4
|   |   ...
|   +--rw ip:ipv6
|   ...
+ ...
```

- subschemas are self-contained and isolated from the top-level schema and other subschemas,
- arbitrary number of nesting levels,
- same modules may be used repeatedly in the top-level schema and/or subschemas.

Components of the Solution

1. YANG library specifies all modules in the top-level schema **including** the module *ietf-yang-schema-mount*.

```
"module": [  
  {  
    "name": "example-logical-devices",  
    "revision": "2016-07-18",  
    "namespace": "urn:example:logical-devices",  
    "conformance-type": "implement"  
  },  
  {  
    "name": "ietf-interfaces",  
    "revision": "2014-05-08",  
    "namespace": "urn:ietf:params:xml:ns:yang:ietf-interfaces",  
    "conformance-type": "implement"  
  },  
  ...  
  {  
    "name": "ietf-yang-schema-mount",  
    "revision": "2016-07-01",  
    "namespace": "urn:ietf:params:xml:ns:yang:ietf-yang-schema-mount",  
    "conformance-type": "implement"  
  }  
]
```

2. **anydata** node(s) containing the mount-point extension statement.

```
module example-logical-devices {
  yang-version 1.1;
  namespace "urn:example:logical-devices";
  prefix exld;


  import ietf-yang-schema-mount {
    prefix yangmnt;
  }

  container logical-devices {
    list logical-device {
      key name;
      leaf name {
        type string;
      }
      anydata root {
        yangmnt:mount-point logical-device;
      }
    }
  }
}
```

3. state data specifying the subschema for each mount point

- a. in place, schema analogical to yang library, or
- b. by referring to a YANG library instance that is mounted under the mount point.

```
{
  "ietf-yang-schema-mount:mount-points": {
    "mount-point": [
      {
        "module": "example-logical-devices",
        "name": "logical-device",
        "modules": {
          "module": [
            {
              "name": "ietf-interface",
              "revision": "2014-05-08",
              "namespace": "urn:ietf:params:xml:ns:yang:ietf-interfaces",
              "conformance-type": "implement"
            },
            ...
          ]
        }
      }
    ]
  }
}
```

 **case a**

Extension Statement mount-point

mount point name

```
yangmnt:mount-point logical-device {  
  yangmnt:mount-yang-library; }  
}
```

← optional, indicates case **b**

Case **b** can be used to define a specific subschema for different entries of a list. This approach doesn't scale because an instance of YANG library has to be present in every list entry.

Alternative approach:

```
anydata ld-foo {  
  when "../type = 'foo'";  
  yangmnt:mount-point logical-device-foo;  
}
```

```
anydata ld-bar {  
  when "../type = 'bar'";  
  yangmnt:mount-point logical-device-bar;  
}
```

Mount Points Only Under **anydata**

Advantages:

- compatible with old clients that don't understand schema mount,
- unique context in which the `mount-point` extension can be used.

Drawbacks:

- adds extra level of schema hierarchy,
- makes the schema less strict – *any* data may be present.

Proposal: Define a capability serving as conformance statement – the server advertising it will accept only data defined by a mounted schema inside an **anydata** instance containing the `mount-point` extension.

Is It What We Want?

The current mechanism should work fine for implementing servers with ad hoc compound schemas, including use cases like peer mount.

However, *it is not a data modelling tool*: the overall schema has to be constructed in an iterative way.

It is unclear how a compound schema involving schema mount could be published, especially with multiple levels of embedding.

DSDL Inspiration

1. External reference pattern in RELAX NG:

```
element logical-devices {  
  external "logical-device.rnc" *  
}
```

Embedded grammars in RELAX NG aren't completely isolated, they can refer to definitions in the parent grammar.

Providing a similar mechanism in YANG would require a new statement.

2. Namespace-based Validation Dispatching Language (NVDL)

A separate meta-schema language is used for splitting the schema of a compound document into fragments based on namespaces, and assigning a schema to each fragment.

Advantages:

- more flexible and modular, existing schemas can be combined in different way,
- schemas expressed in different schema languages can be used in the same compound schema.

YSDL (draft-1hotka-netmod-ysdl-00, expired) tried to adapt this approach to YANG.