YANG Schema Mount

draft-ietf-schema-mount-04

Martin Björklund
〈mbj@tail-f.com〉

Ladislav Lhotka
〈lhotka@nic.cz〉

28 March 2017
Main Changes since -03

- Mount points can be under **container** and **list**, not **anydata**.
  YANG: contents of anydata instances have to be treated as blobs.
- Leafrefs and instance identifiers in a mounted schema can be redirected to the parent schema
- Read-only mounts
### Mount Point

Indicated in the schema tree via `mount-point` extension, two meanings:

<table>
<thead>
<tr>
<th>use-schema</th>
<th>inline</th>
</tr>
</thead>
</table>
| container foo {  
  yangmnt:mount-point froot;  
} | location in the *schema* where new schema nodes will be placed (similar to augment) | location in the *instance* where YANG library (+ other data) will be placed |
| list bar {  
  key id;  
  yangmnt:mount-point broot;  
} | same as above | YANG library data will be placed in *every entry* of the list (needn’t be identical) |
Parent Schema References

Generally, XPath expressions in the mounted schema cannot refer to nodes outside the “mount jail”. This is sometimes too restrictive.

Typical use case: network instances [draft-ietf-rtgwg-ni-model]:

**Simple router**

```
+--rw if:interfaces
+--rw rt:routing
...
```

**Router with VRF**

```
+--rw if:interfaces
+--rw network-instances
   +--rw network-instance* [name]
      +-- rw root
         +--rw rt:routing
         ...
```

We want to use the same *ietf-routing* in both cases but routing configuration refers to interfaces that are outside the mounted schema in the VRF case.
Current Solution

```plaintext
+-ro mount-point* [module name]
  |  +-ro module
  |  |  +-ro name
  |  |  +-ro config?
  |  |  +-ro (schema-ref)?
  |  |  |  +-(inline)
  |  |  |  |  |  +-ro inline?
  |  |  |  +-(use-schema)
  |  |  |  |  |  +-ro use-schema* [name]
  |  |  |  |  |  |  +-ro name
  |  |  |  |  |  |  +-ro when?
  |  |  |  |  |  |  +-ro parent-reference*
  |  |  +-ro schema* [name]
  ...
```

State data for NI example:

```json
"mount-point": [
  {
    "module": "ietf-network-instance",
    "name": "root",
    "use-schema": [
      {
        "name": "network-instance",
        "parent-reference": [
          "ietf-routing",
          ...
        ]
      }
  }
  ...
]
```

```
"schema": [
  {
    "name": "network-instance",
    "module": [
      {
        "name": "ietf-routing",
        ...
      }
    ],
    ...
  }
]```

Only available for the **use-schema** case.

Redirection to the parent schema works for absolute leafref paths and instance-identifiers, **not** for general XPath.

← top-level modules for which the redirection will be applied
Read-Only Mount

In some scenarios (split management) it is useful to provide mounted data at the parent schema level only for reading.

Example – LNE model [draft-ietf-rtgwg-lne-model]:

```
"mount-point": [
  {
    "module": "ietf-logical-network-element",
    "name": "root",
    "config": false,
    "inline": [null]
  }
]
```
Important Open Issues

1. Reference mount points with schema node identifiers
2. Restrictions on parent schema references
3. Mount points inside keyless lists
4. Design-time mounts
#1: Mount Point References

Use schema node identifiers instead of `mount-point` extension?

```
"ietf-yang-schema-mount:schema-mounts": {
  "namespace": [  
    { "prefix": "ni",  
      "ns-uri": "urn:ietf:params:xml:ns:yang:ietf-network-instance"  
    },

  ]

  "mount-point": [  
      "parent-reference": ["ietf-interfaces"],  
      "use-schema": [  
        { "name": "ni-schema" }  
      ]
    ],

  ],

  ...
```

Possible solution:

- **use** `mount-point` extension for the *inline* case (no need for additional state data)

- integrate the *use-schema* case (with schema node identifiers) into the new YANG library that is being proposed by [draft-ietf-netmod-revised-datasstores].
#2: Restrictions on parent schema references

Currently allowed only for absolute leafref paths and instance identifiers – is it sufficient?

Extending it to all XPath expressions would seriously complicate their evaluation. For example, if we have as in the NI model

"parent-reference": ["ietf-interfaces", ...]

would it also apply to "/ip:ipv4"?

Alternative solution – indicate the parent reference explicitly

- using a new XPath function `parent-root()` that evaluates to the parent root node, or global root node if used in the top-level schema (example: "parent-root()//ip:ipv4");

- using an XPath variable, and then setting appropriate values in the parent and mounted schemas (example: "$root//ip:ipv4").
#3: Mount Points Inside Keyless Lists

Schema mount turns RPCs and notifications into actions/notifications connected to the mount point – not permitted by RFC 7950 if one of the ancestors of mount point is a list without a key statement.

**Proposal:** Require that a either

- the mount point have no keyless lists among its ancestors, or
- the mounted schema has defines no RPCs, actions or notifications.
#4: Design-Time Mounts

Allow for specifying mounted modules along with the definition of the mount point.

```yang
yangmnt:mount-point root {
    yangmnt:mount-module "ietf-foo";
    ...
}
```

Problems:

- If `ietf-foo` is augmented in the top-level schema, does the augment apply also to the mounted module?
- More information is needed: revision of `ietf-foo`, supported features etc.