draft-vivek-simap-external-relationship-00

Vivekananda Boudia
vivekananda.boudia@insa-lyon.fr

Pierre Francois
pierre.francois@insa-lyon.fr

Sherif Mostafa
sherif.mostafa@huawei.com

Agenda

- Why?
- External relationship yang model
- Q&A

Use Cases

- (See the long list of use-cases from interim meeting)
- Many scenarios where information is not maintained within a SIMAP itself
  - State
  - What-if's
  - Replay's
- We need to describe how to link network elements to external information
  - Already used in the SIMAP hackathon effort

External relationship yang model

- Template
    - Requests + Parameters → HOW
    - Type → WHAT
- Relation
  (Net-element → Templates)
    - Supported Request Types
    - Parameter Values

Template - Request

Request describes how to execute a query to obtain external information

```
grouping netconf-request {
    description
      "netconf request";
    leaf xpath {
      type string;
      description
        "netconf xpath for request";
    }
  }
```

```
list request {
  key "request-type";
  description
    "request list";
  leaf request-type {
    type identityref {
      base request-type;
    }
    description
      "request type";
  }
  container request-builder {
    description
      "request container that allows users to retrieve data
       parameters must be enclosed in brackets.";
  }
}

```

```
augment "/template/template/request/request-builder" {
  when "derived-from-or-self(../request-type, 'NETCONF')";
  uses netconf-request;
  description
    "adding netconf request to possible request";
}
```

Template - Parameter

Parameters list what needs to be provided to get the information

```
        }
        list parameter {
          key "param-id";
          description
            "list of parameter used by request";
          leaf param-id {
            type inet:uri;
            description
              "uniquely identifies a parameter";
          }
          leaf description {
            type string;
            description
              "parameter description";
          }
        }
```

Template - Type

Type characterizes what information is queried with the template

```
identity CONFIG {
  base template-type;
  description
    "template used to retrieve configuration data";
}

identity STATE {
  base template-type;
  description
    "template used to retrieve operational state data";
}
```

```
container template-type {
  description
    "template type
     used for filtering";
  leaf base {
    type identityref {
      base template-type;
    }
    description
      "template base
       used for filtering";
  }
  leaf is-historical {
    type boolean;
    description
      "check if template is used to get historical data or not
       used for filtering";
  }
  leaf-list extra-label {
    type identityref {
      base extra-label;
    }
    description
      "extra label
       used for filtering";
  }
}
```

Relation

We use

- network element ref
- template ref

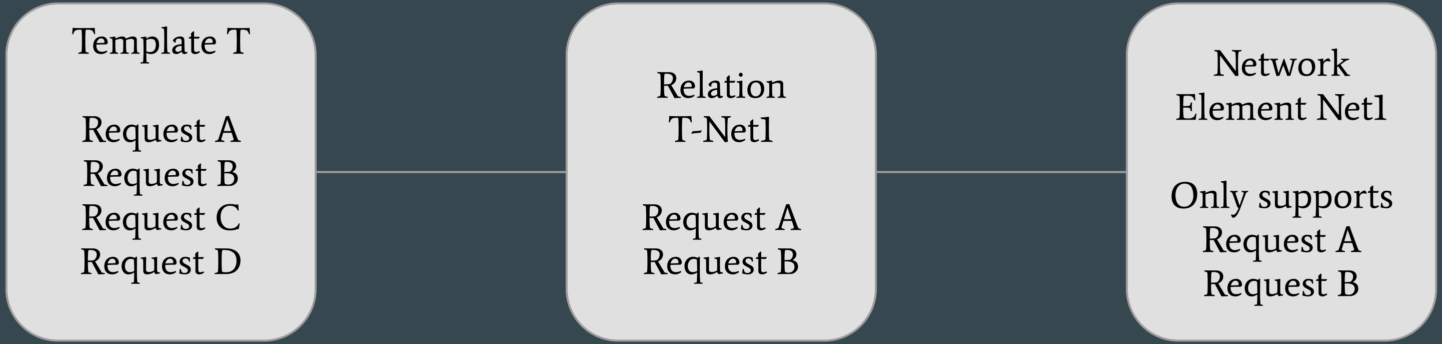
to associate a RFC8345 element to a template

```
}
choice network-element-ref {
  description
    "linking to RFC8345";
  leaf network-ref {
    type leafref {
      path "/nw:networks/nw:network/nw:network-id";
    }
    description
      "linking to network";
  }
  leaf node-ref {
    type leafref {
      path "/nw:networks/nw:network/nw:node/nw:node-id";
    }
    description
      "linking to node";
  }
  leaf link-ref {
    type leafref {
      path "/nw:networks/nw:network/nt:link/nt:link-id";
    }
    description
      "linking to link";
  }
  leaf tp-ref {
    type leafref {
      path "/nw:networks/nw:network/nw:node/nt:termination-point/nt:tp-id";
    }
    description
      "linking to termination point";
  }
}
leaf template-ref {
  type leafref {
    path "/dr-tmp:template/dr-tmp:template/dr-tmp:template-id";
  }
  description
    "reference to template";
}
```

Relation - Request Type Supported

Relations are used to specify which request is supported by which network element

```
      leaf-list request-type-supported {
        type identityref {
          base dr-tmp:request-type;
        }
        description
          "template is generic and may include requests that are not supported by the network element
           here, we specify the types of requests that the network element supports
           if network element supports all template ALL-REQUEST may be used";
      }
```



Relation - Parameter Value

Value of the parameter used by the request

```
list parameter-value {
  key "param-ref request-type";
  description
    "parameter value for network element";
  leaf param-ref {
    type leafref {
      path "/dr-tmp:template/dr-tmp:template/dr-tmp:parameter/dr-tmp:param-id";
    }
    description
      "reference to template parameter";
  }
  leaf request-type {
    type identityref {
      base dr-tmp:request-type;
    }
    description
      "value can be different depending on the request";
  }
  leaf value {
    type string;
    description
      "value of the parameter";
  }
}
```

Thanks

- Questions / Comments?
- WG Doc?