Yang Templates (Cisco)

Robert Wills

Overview

Defining a template

- A template can be defined for any fragment of the data model
- Templates are stored in the configuration datastore
- A template can use regexes to specify which nodes it can be applied to

Applying a template

- A template can be applied to the data model at any point in the hierarchy using a new attribute
- Template application “cascades down” to children

Defining a template

```
module yang-template {
  prefix yt;

  container templates {
    list template {
      key name;
      leaf name { type string; }
      container config { type anyxml; }              // template content goes here
    }
  }
}
```

Defining a template

```
<yt:templates>
  <yt:template>                                          // list of templates
    <yt:name>set_physical_mtu</yt:name>                  // template name
    <yt:config>




    </yt:config>
  </yt:template>
</yt:templates>
```

Defining a template

```
<yt:templates>
  <yt:template>                                          // list of templates
    <yt:name>set_physical_mtu</yt:name>                  // template name
    <yt:config>
      <interfaces>
        <interface>
          <name>GigabitEthernet.*</name>                 // regex specifies which data nodes can
apply the template
          <ipv4>
            <mtu>1400</mtu>                              // for instances applying this template,
set this leaf
          </ipv4>
        </interface>
      </interfaces>
    </yt:config>
  </yt:template>
</yt:templates>
```

Note: Defining a template does not apply it!

Applying a template

```
<interfaces>
  <interface yt:apply-templates="set_physical_mtu">
    <name>GigabitEthernet0/0/0/0</name>
  </interface>
  <interface>
    <name>GigabitEthernet0/0/0/1</name>
  </interface>
</interfaces>
```

```
<interfaces>
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
    <mtu>1400</mtu>
  </interface>
  <interface>
    <name>GigabitEthernet0/0/0/1</name>
  </interface>
</interfaces>
```

Applying a template

```
<interfaces yt:apply-templates="set_physical_mtu">
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
  </interface>
  <interface>
    <name>GigabitEthernet0/0/0/1</name>
  </interface>
</interfaces>
```

```
<interfaces>
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
    <mtu>1400</mtu>
  </interface>
  <interface>
    <name>GigabitEthernet0/0/0/1</name>
    <mtu>1400</mtu>
  </interface>
</interfaces>
```

Applying a template

```
<interfaces yt:apply-templates="set_physical_mtu">
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
  </interface>
  <interface>
    <name>Loopback0</name>
  </interface>
</interfaces>
```

```
<interfaces>
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
    <mtu>1400</mtu>
  </interface>
  <interface>
    <name>Loopback0</name>
    // mtu config item is not here, because the interface name does not match the template's regex
  </interface>
</interfaces>
```

Excluding a template

```
<interfaces yt:apply-templates="set_physical_mtu">
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
  </interface>
  <interface yt:exclude-templates="set_physical_mtu">
    <name>GigabitEthernet0/0/0/1</name>
  </interface>
</interfaces>
```

```
<interfaces>
  <interface>
    <name>GigabitEthernet0/0/0/0</name>
    <mtu>1400</mtu>
  </interface>
  <interface>
    <name>GigabitEthernet0/0/0/1</name>
    // mtu config item is not here, because the inherited template was excluded
  </interface>
</interfaces>
```

draft-wills-netmod-yang-templates-00