Notification Capabilities Model Extension for Data Node Tag Capability Support
draft-tao-netconf-notification-node-tag-capabilities-00

Ran Tao
Bo Wu
Background

- Notification capability model allows a client to discover YANG-Push related server capabilities such as;
  - Maximum number of objects that can be sent in an update
  - Supported (reporting) periods for periodic subscriptions

- It is defined in [l-D.netconf-notification-capabilities]

- Without using notification capability, it leads to unexpected failure or additional message exchange for NETCONF clients to discover data models supported by a NETCONF server.

- When the state of all subscriptions of a particular Subscriber to be fetched is huge, filtering queries of operational state on a server based on server capabilities can greatly reduce the amount of data to be streamed out to the destination.
Why this draft

YANG push telemetry provide a mechanism to subscribe to and select operational state data objects based on selection filter [RFC8641].

- However there is no document to discuss how selection filter is specified.
- Without preconfigured selection filter by human intervening or service logic defined by the application

It is hard for NETCONF clients to automatically select which data objects are of interest or which data objects should be subscribed, e.g.,

- identify a set of objects which have a common characteristic,
- collect specific object type nodes.
Data Node Tag Capability Proposal

- This document provides a dynamic way to specify selection filters:
  - Tag characteristics data object and instruct the client or subscriber to select operation state data based on characteristics data object.
- Data Node Tag Capability Model provides self-explanation information for data objects of interests.

- Augment Notification Capability model [I-D.netconf-notification-capabilities]
  - Specification of which object type nodes they can push to the target recipient.
  - Specification of which group of data nodes they can push to the target recipient.

- These self-explanation information for data objects can be used to:
  - Further filter queries of operational state on a server based on server capabilities.
  - Correlate data node across models that share common characteristics or are of same object types.
  - Reduce the amount of data to be streamed out to the destination.
Data Node Tag Capability Example

1. Tag data node in each device model to indicate data that has common characteristic. Tagging information can be stored in network element or controller/website.

2. Automatically advertiseTag from live server in device or from offline server (node-tag: Performance-metric group-id: A).

3. Subscribe interested data based on self explanation tag indication and xpath of data node.

<table>
<thead>
<tr>
<th>Device Name</th>
<th>telemetry data node</th>
<th>xpath</th>
<th>Self-explanation tag</th>
<th>Statistics operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device A</td>
<td>Node A</td>
<td>foo/A</td>
<td>Performance metric</td>
<td>min</td>
</tr>
<tr>
<td>Device A</td>
<td>Node C</td>
<td>foo/C</td>
<td>Performance metric</td>
<td>average</td>
</tr>
<tr>
<td>Device B</td>
<td>Node E</td>
<td>bar/E</td>
<td>Performance metric</td>
<td>max</td>
</tr>
<tr>
<td>Device C</td>
<td>Node H</td>
<td>car/H</td>
<td>Performance metric</td>
<td>min</td>
</tr>
</tbody>
</table>

I am interested in Node A, C in Module Foo, Node E in Module bar.

NMS

Module foo {
  Node A //performance metric;min
  Node B
  Node C//performance metric;average
}

Module bar {
  Node E //performance metric;max
  Node F
  Node G
}

Module car {
  Node H //performance metric;min
  Node I
  Node J
}
Next Step

• Decide whether this needs to be documented as RFC by NETCONF WG
• Any comments and input are welcome!