

YANG PUSH Based Generalized Network Control Automation

draft-bryskin-netconf-automation-framework-00

Igor Bryskin (Huawei Technologies)

Xufeng Liu (Jabil)

Alex Clemm (Huawei Technologies)

Henk Birkholz (Fraunhofer SIT)

Tianran Zhou (Huawei Technologies)

Evolution of YANG Based Network Automation

- **"Custom Subscription to Event Notifications"** model :
 - allows for a client to subscribe to unsolicited event notifications defined by supported YANG models;
- **"Subscribing to YANG datastore push updates"** model:
 - allows for a client to define subscribable events and contents of event notifications as target-trigger-notify triplets
- **"Smart filters for Push Updates"** model:
 - allows for a client to filter event triggers and notifications on push object values and their change history, focuses on "outliers"

Objectives of YANG PUSH Based Generalized Network Control Automation

1) To generalize **target-trigger-notify** concept into **event-condition-action** concept, where:

event - a particular change in the network state explicitly defined by one of the YANG models supported by the network or implicitly defined by the client, which is constantly monitored by the network;

condition - a logical expression that is evaluated only once after the associated event is detected;

action - an operation to be carried out by the network when the associated event is detected and the associated condition is met

Objectives of YANG PUSH Based Generalized Network Control Automation

- 2) To provide for a client a capability to configure the **event-condition-action** triplets as **policy rules** ahead of time or/and during network operations

Generalized Action

- Send **notification**
- Perform immediate network **reconfiguration** (e.g. modify one or more attributes of one or more CONFIG=TRUE data store nodes);
- **Schedule** one time or periodic reconfiguration in the future;
- **Call RPC** defined by one of the YANG models supported by the network (e.g. call network's path computer to evaluate whether an alternative/more optimal path is available for a given connection)
- **Link/unlink** data store dynamic sub-trees;
- Etc.

Relationship with Policy Framework

- The framework should work autonomously
- The framework should fit well within a higher level policy framework, with the latter possibly providing a greater level of automation:
 - multiple **micro-conditions** could be combined into a single **macro-condition** via a number of logical operations;
 - multiple **micro-actions** could be combined into a single **transaction** with a possibility of specifying policies with respect to handling errors/exceptions of each of the transaction components

Framework Benefits

- lower latency, faster responsiveness of the network to various events/conditions;
- better scale (e.g. the client may control more networks because it does not have to monitor/micro-manage any of them);
- CPU and bandwidth savings due to the reduced amount of communication between the client and the network
- The client can take itself out of the network control loop, change its role from being network's "micro-manager" to being network's "police officer", who interferes into network operations only in exceptional/unpredicted situations

Next steps

- Solicit feedbacks from Netconf WG
- Design Generalized Network Control Automation YANG model(s)