Generalized Network Control
Automation YANG Model
draft-bryskin-netconf-automation-yang-02

Igor Bryskin (Huawei Technologies)
Xufeng Liu (Volta Networks)
Alex Clemm (Huawei Technologies)
Henk Birkholz (Fraunhofer SIT)
Tianran Zhou (Huawei Technologies)
Objectives

• **Purpose**: to manipulate network close loop automation via configuration of standardized Event-Condition-Action (ECA) containers

• **ECA** – a set of NETCONF style requests/primitives (e.g. get data, edit-config, call-rpc, etc), whose execution on the server is triggered by a specified event, and whose order of execution is conditioned by current and/or historical network states and/or their derivatives

• Explicit **non-goal**: introducing a new interpreter/language/scripting environment
ECAs, when and why

• Reaction to events could be articulated to the network server in advance
• To enhance network responsiveness to events
• To improve scalability of network control
• To configure on the server programmable by a client logic
Policy Variables

• Policy Variable (PV) is an ECA state, i.e. a structure to keep results of the ECA execution for immediate or future use.

• PV types: global (shared between ECAs), local (ECA scope, static or dynamic).

• PV content structure:
  - of a common type (e.g. integer, uint64, etc.)
  - Or
  - of an existing YANG node pointed by XPath (e.g. TE_Topologies/links/te_link)
What could be done with PVs?

• **read** from/**write** to YANG data store
• Used as input/output when calling **YANG RPCs**
• Used to generate **notification** messages;
• Used as input/output for **function** calls, for example `Fmult(a, 0.75)` to calculate `0.75*a`
• Used in XPath expressions with PVs referred to by their respective positions in the YANG tree
ECA Events

• Subscribable events:
  - explicitly defined by YANG modules
  - YANG Push or/and smart filter subscriptions

• Timers
ECA Conditions

• Logical expressions with YANG data store nodes and/or PVs
• A condition could be configured as:
  - a single XPath expression
  - a hierarchy of comparisons and logical combinations of thereof
    (e.g. \((X == Y \ | | A<B) \ & \ & (C<=D \ | | E>F)\))
ECA Actions

• NETCONF style primitives:
  - get data, edit-config, etc.
  - calling YANG defined **RPCs** (e.g. TE_TunnelPathComputation RPC defined by YANG TE Tunnel model)
  - sending **notification** messages to the client
  - adding/removing event notification **subscriptions**

• Starting/stopping **timers**

• Calling other **ECAs**

• Performing **operations on PVs** (e.g. function calls)
ECA Structure

- **Event** name
- List of local PVs
- **Normal** Condition-Action list
- **Cleanup** Condition-Action list (to undo actions from the normal Condition-Action list in case one of the normal actions was rejected by the server)
Changes since IETF101

• A mechanism is introduced to define ECA events via YANG PUSH subscriptions

• A structure is defined to correlate by the client ECA notifications with corresponding events
Defining ECA Events via YANG PUSH Subscriptions

• Client:
  - configures a PUSH subscription
  - configures an ECA event specifying the PUSH subscription name
Defining ECA Events via YANG PUSH Subscriptions

• Server (at the moment of event configuration):
  - registers the event interpreting the referred PUSH subscription trigger as event firing trigger
  - uses the PUSH subscription filters to auto-configure the event’s ECA input in the form of ECA’s local PVs
• Server (at the moment of even firing):
  - copies data store states pointed by the PUSH subscription filters into ECA’s auto-configured local PVs and triggers the ECA’s Condition-Action chain execution
ECA Events and Notifications Correlation

- ECA notification is the only ECA action that interacts directly with the client
- Multiple ECAs could be triggered simultaneously, each of which potentially generating multiple semantically different one-time and/or repetitive notifications
- Every ECA notification message includes mandatorily the event name and event-scope unique notification name to facilitate for the client event-notification correlation
Next Steps

• Solicit more discussions and feedback from WG