Subscribing to YANG datastore push updates draft-ietf-netconf-yang-push-02

NETMOD WG
IETF #95 Buenos Aires
4-April-2015

Alexander Clemm
Alberto Gonzalez Prieto
Eric Voit
Ambika Prasad Tripathy
Einar Nilsen-Nygaard

<alex|albertgo|evoit|ambripa|einarnn@cisco.com>

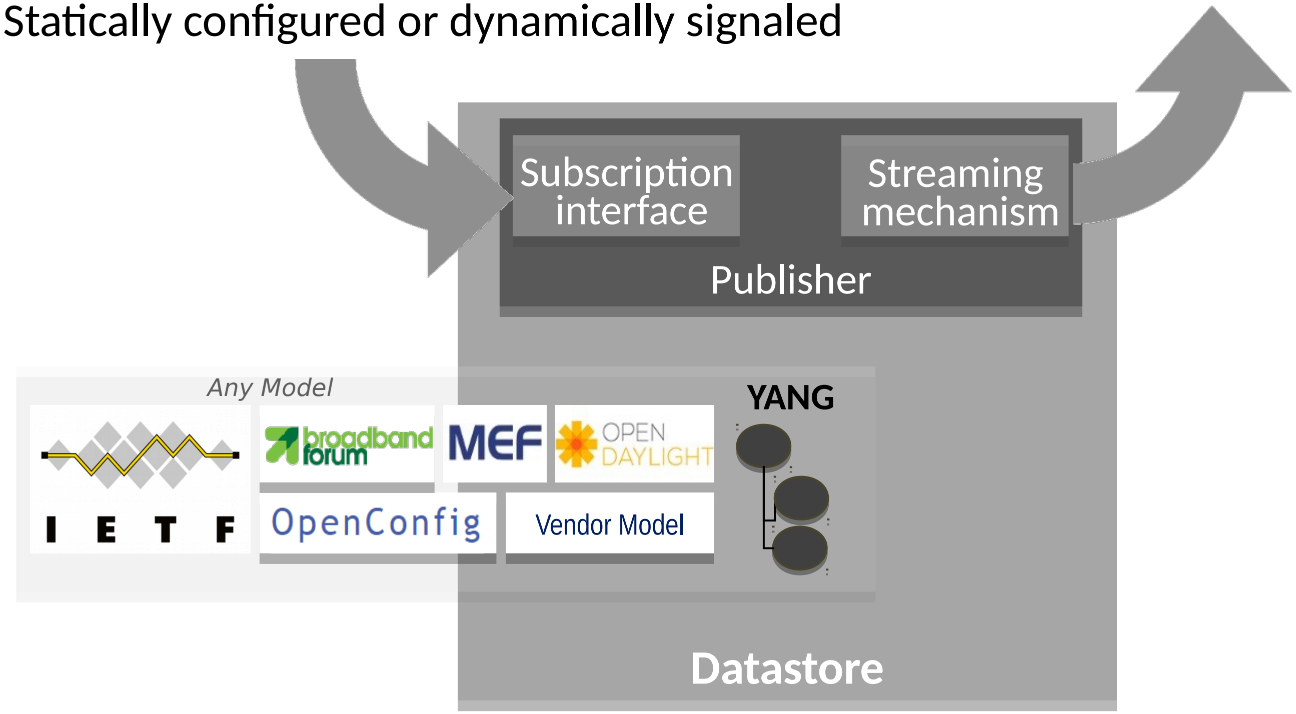
draft-ietf-netconf-yang-push

Streaming of updates

- Customized to recipient
- On-change or Periodic

Subscribing to updates

- Any YANG subtree on device
- Statically configured or dynamically signaled



Differentiating NETCONF Notification & YANG Push

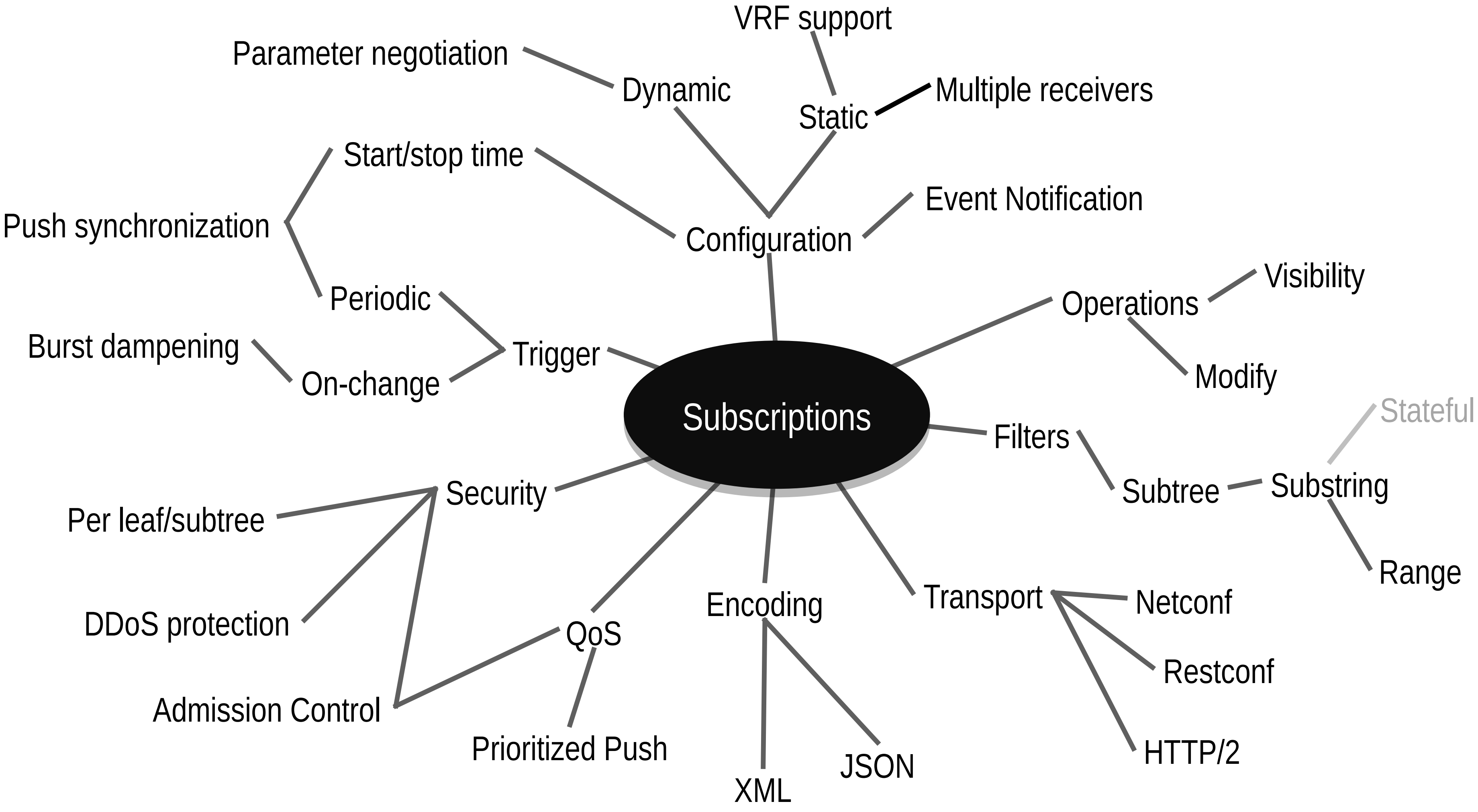
complimentary

What you need	Consume a stream of Publisher generated **messages at the cadence determined by the Publisher**	Consume a stream of Publisher generated **YANG data updates at a cadence negotiated with the Subscriber**
What to Use	NETCONF Event Notifications	YANG Push
Drafts / RFCs	• RFC 5277 • draft-gonzalez-netconf-5277bis	draft-ietf-i2rs-pub-sub-requirements **draft-ietf-netconf-yang-push** draft-voit-netconf-restconf-yang-push

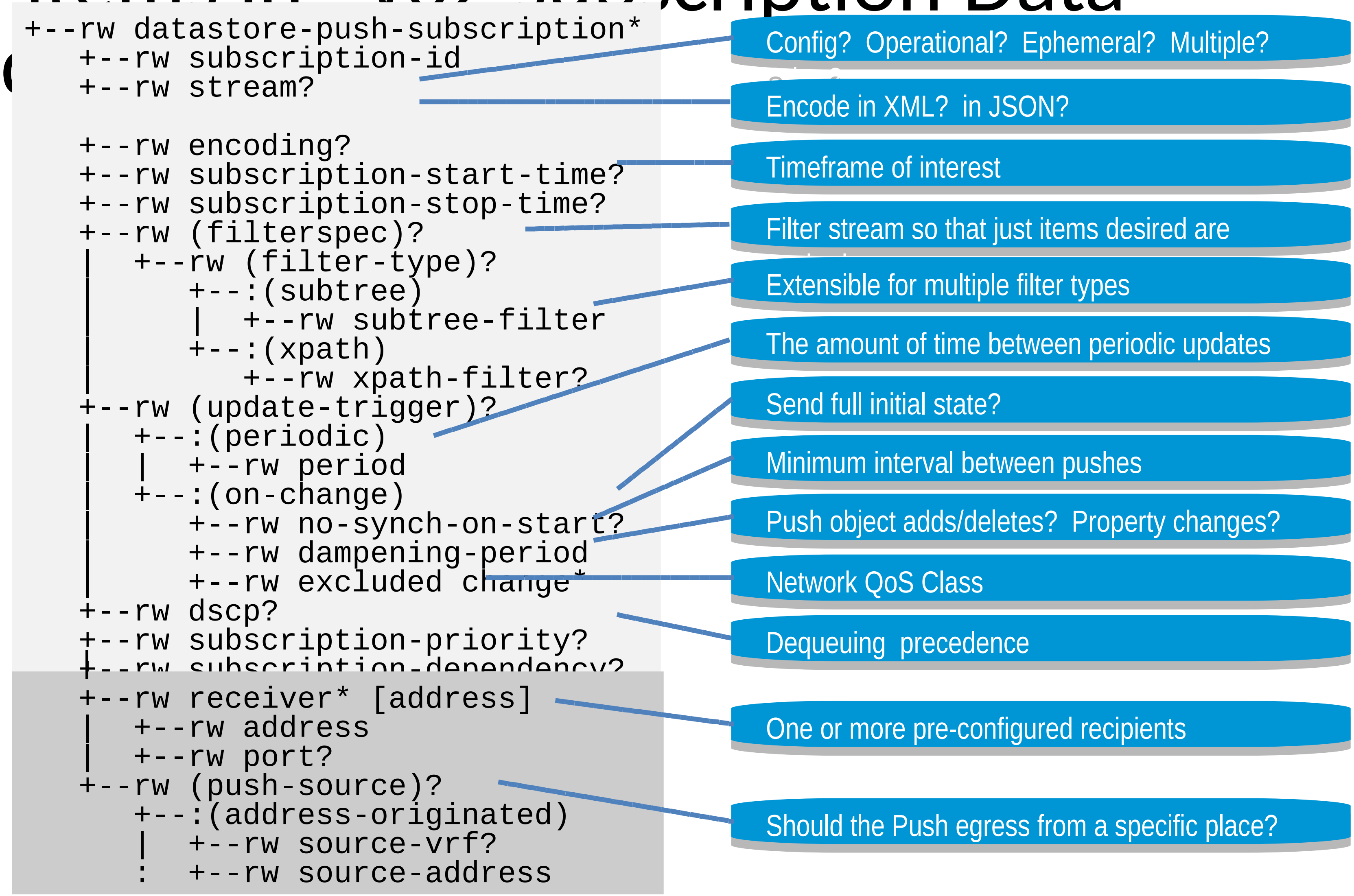
For discussion at NETCONF (Thursday)

Focus today

Support requirements of draft-ietf-i2rs-pub-sub-requirements



draft-ietf-netconf-yang-push
Key items in –v02 subscription Data Mo



```
+--rw datastore-push-subscription*
   +--rw subscription-id
   +--rw stream?

   +--rw encoding?
   +--rw subscription-start-time?
   +--rw subscription-stop-time?
   +--rw (filterspec)?
   |  +--rw (filter-type)?
   |     +--:(subtree)
   |     |  +--rw subtree-filter
   |     +--:(xpath)
   |        +--rw xpath-filter?
   +--rw (update-trigger)?
   |  +--:(periodic)
   |  |  +--rw period
   |  +--:(on-change)
   |     +--rw no-synch-on-start?
   |     +--rw dampening-period
   |     +--rw excluded change*
   +--rw dscp?
   +--rw subscription-priority?
   +--rw subscription-dependency?
   +--rw receiver* [address]
   |  +--rw address
   |  +--rw port?
   +--rw (push-source)?
      +--:(address-originated)
      |  +--rw source-vrf?
      :  +--rw source-address
```

Config? Operational? Ephemeral? Multiple?

Encode in XML? in JSON?

Timeframe of interest

Filter stream so that just items desired are

Extensible for multiple filter types

The amount of time between periodic updates

Send full initial state?

Minimum interval between pushes

Push object adds/deletes? Property changes?

Network QoS Class

Dequeuing precedence

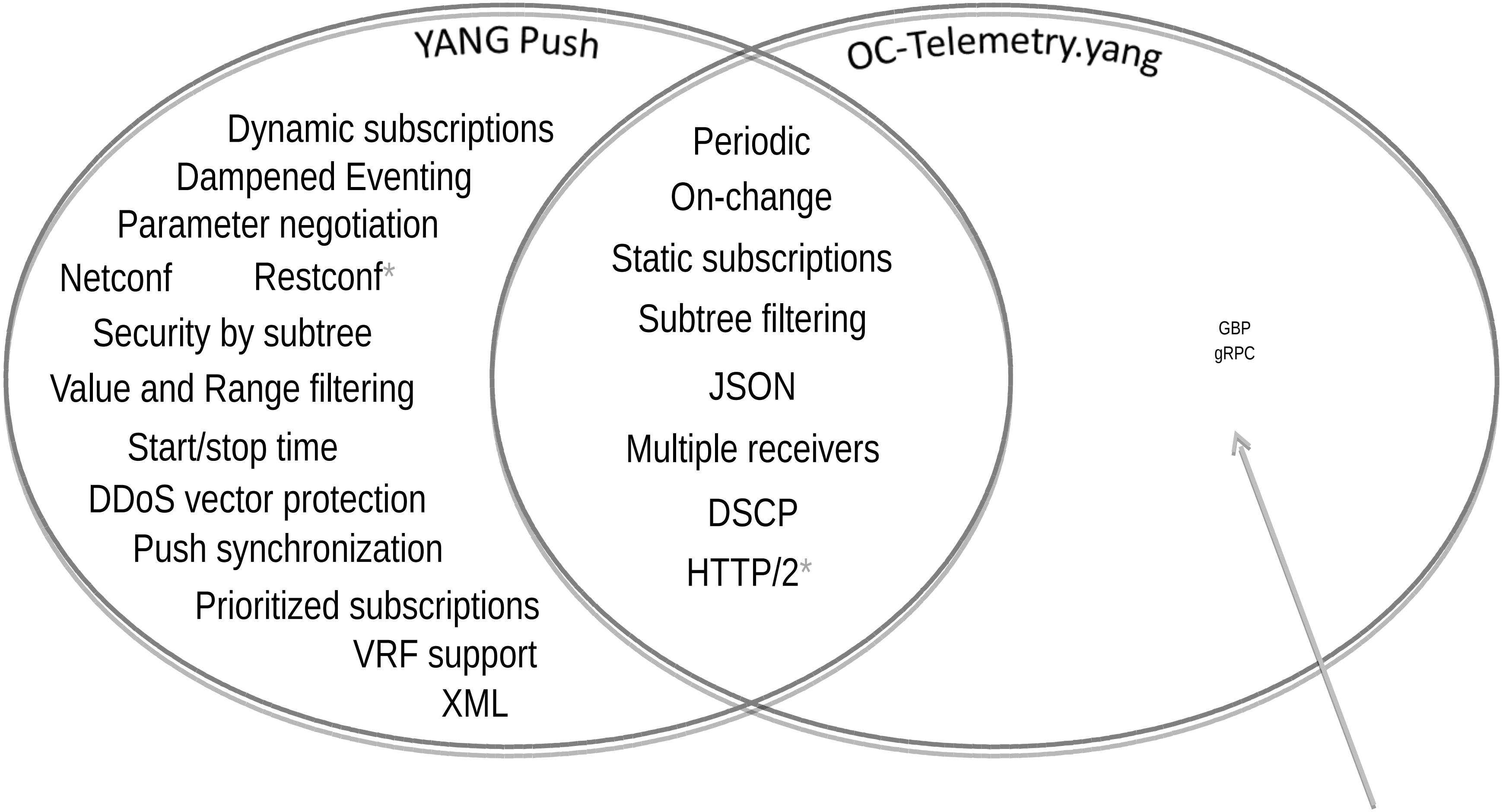
One or more pre-configured recipients

Should the Push egress from a specific place?

Signaled
(Dynamic)
Subscriptions

+ also for
Configured
(Static)
Subscriptions

YANG Push draft Support, and Gaps



YANG Push

OC-Telemetry.yang

Dynamic subscriptions

Dampened Eventing

Parameter negotiation

Netconf Restconf*

Security by subtree

Value and Range filtering

Start/stop time

DDoS vector protection

Push synchronization

Prioritized subscriptions

VRF support

XML

Periodic

On-change

Static subscriptions

Subtree filtering

JSON

Multiple receivers

DSCP

HTTP/2*

GBP

gRPC

Open Question: How do we support market requested, non-IETF technologies?

* via draft-voit-netconf-restconf-yang-push

YANG Push results at IETF95 Hackathon

Objective

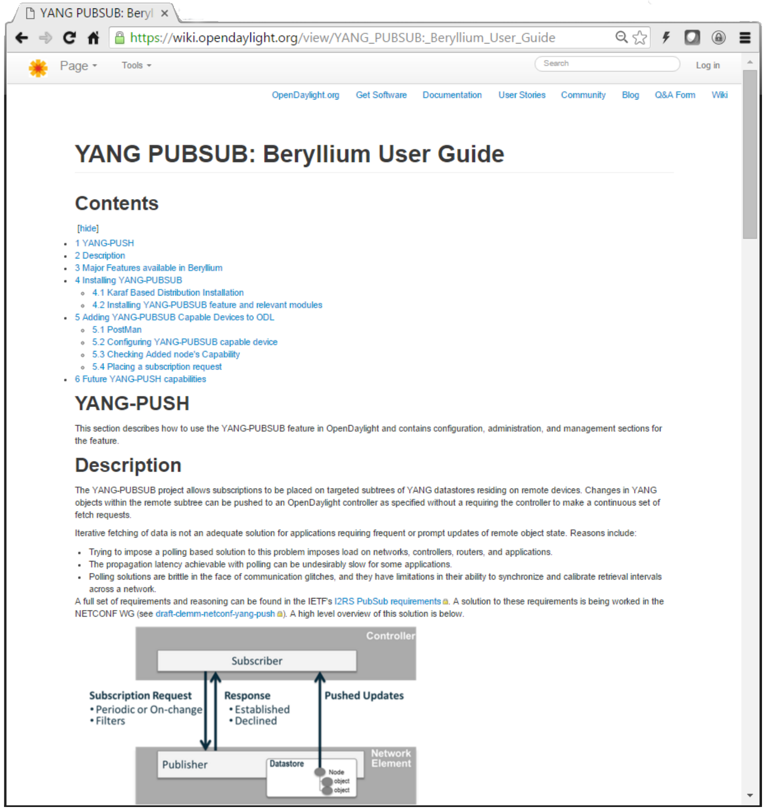
- Extend OpenDaylight’s YANG Push Client

Coded / Demonstrated

- Create Subscription (via OpenDaylight)
- Delete Subscription (via NETCONF session)
- On-change subscription (via NETCONF session)

Interested?

- Will Show again this Thursday night’s “Bits & Bytes”



YANG PUBSUB: Beryllium User Guide

Contents

[hide]

YANG-PUSH

This section describes how to use the YANG-PUBSUB feature in OpenDaylight and contains configuration, administration, and management sections for the feature.

Description

The YANG-PUBSUB project allows subscriptions to be placed on targeted subtrees of YANG datastores residing on remote devices. Changes in YANG objects within the remote subtree can be pushed to an OpenDaylight controller as specified without a requiring the controller to make a continuous set of fetch requests.

Iterative fetching of data is not an adequate solution for applications requiring frequent or prompt updates of remote object state. Reasons include:

- Trying to impose a polling based solution to this problem imposes load on networks, controllers, routers, and applications.
- The propagation latency achievable with polling can be undesirably slow for some applications.
- Polling solutions are brittle in the face of communication glitches, and they have limitations in their ability to synchronize and calibrate retrieval intervals across a network.

A full set of requirements and reasoning can be found in the IETF's I2RS PubSub requirements. A solution to these requirements is being worked in the NETCONF WG (see draft-clemm-netconf-yang-push). A high level overview of this solution is below.

Questions?

- Also will do a deeper dive in NETCONF on Thursday….

Thank you!

Model overview: streams + filters

```
module: ietf-yang-push
   +--ro update-streams
   |  +--ro update-stream*   update-stream
   +--rw filters
   |  +--rw filter* [filter-id]
   |     +--rw filter-id         filter-id
   |     +--rw (filter-type)?
   |        +--:(subtree)
   |        |  +--rw subtree-filter
   |        +--:(xpath)
   |        |  +--rw xpath-filter?     yang:xpath1.0
   |        +--:(rfc5277)
   |           +--rw filter
```

Model overview: Subscription configuration

```
+--rw subscription-config {configured-subscriptions}?
   |  +--rw yang-push-subscription* [subscription-id]
   |     +--rw subscription-id            subscription-id
   |     +--rw stream?                    update-stream
   |     +--rw encoding?                  encoding
   |     +--rw subscription-start-time?   yang:date-and-time
   |     +--rw subscription-stop-time?    yang:date-and-time
   |     +--rw (filterspec)?
   |     |  +--:(inline)
   |     |  |  +--rw (filter-type)?
   |     |  |     +--:(subtree)
   |     |  |     |  +--rw subtree-filter
   |     |  |     +--:(xpath)
   |     |  |     |  +--rw xpath-filter?              yang:xpath1.0
   |     |  |     +--:(rfc5277)
   |     |  |        +--rw filter
   |     |  +--:(by-reference)
   |     |     +--rw filter-ref?                filter-ref
```

Optional feature, only applies to static subscriptions

Model overview:
Subscription configuration (contd)

```
+--rw subscription-config {configured-subscriptions}?
   |  +--rw yang-push-subscription* [subscription-id]
…
   |     +--rw (update-trigger)?
   |     |  +--:(periodic)
   |     |  |  +--rw period                     yang:timeticks
   |     |  +--:(on-change) {on-change}?
   |     |     +--rw no-synch-on-start?         empty
   |     |     +--rw dampening-period           yang:timeticks
   |     |     +--rw excluded-change*           change-type
   |     +--rw receiver* [address]
   |     |  +--rw address     inet:host
   |     |  +--rw port?       inet:port-number
   |     |  +--rw protocol?   transport-protocol
   |     +--rw (push-source)?
   |     |  +--:(interface-originated)
   |     |  |  +--rw source-interface?          if:interface-ref
   |     |  +--:(address-originated)
   |     |     +--rw source-vrf?                uint32
   |     |     +--rw source-address             inet:ip-address-no-zone
   |     +--rw dscp?               inet:dscp {configured-subscriptions}?
   |     +--rw subscription-priority?     uint8
   |     +--rw subscription-dependency?   String
```

Note: items in blue are new or modified since -00

Model overview:
Subscription monitoring

```
+--ro subscriptions
     +--ro yang-push-subscription* [subscription-id]
        +--ro subscription-id            subscription-id
        +--ro configured-subscription?empty {configured-subscriptions}?
        +--ro subscription-status?       identityref
        +--ro stream?                    update-stream
        +--ro encoding?                  encoding
        +--ro subscription-start-time?   yang:date-and-time
        +--ro subscription-stop-time?    yang:date-and-time
        +--ro (filterspec)?
        |  +--:(inline)
             …
        |  +--:(by-reference)
        |     +--ro filter-ref?                 filter-ref +--ro (update-trigger)?
        |  +--:(periodic)
        |  |  +--ro period                      yang:timeticks
        |  +--:(on-change) {on-change}?
        |     +--ro no-synch-on-start?          empty
        |     +--ro dampening-period            yang:timeticks
        |     +--ro excluded-change*            change-type
        +--ro receiver* [address]
        |  +--ro address     inet:host
        |  +--ro port?       inet:port-number
        |  +--ro protocol?   transport-protocol
        +--ro (push-source)?
        |  +--:(interface-originated)
        |  |  +--ro source-interface?          if:interface-ref
        |  +--:(address-originated)
        |     +--ro source-vrf?                uint32
        |     +--ro source-address             inet:ip-address-no-zone
        +--ro dscp?                inet:dscp {configured-subscriptions}?
        +--ro subscription-priority?     uint8
        +--ro subscription-dependency?   string
```

- Applies to both dynamic and static subscriptions
- Mirrors subscription parameters from RPC or config

Note: items in blue are new or modified since -00

Model overview: RPCs (input)

```
+---x establish-subscription
  |  +---w input
  |  |  +---w stream?                    update-stream
  |  |  +---w encoding?                  encoding
  |  |  +---w subscription-start-time?   yang:date-and-time
  |  |  +---w subscription-stop-time?    yang:date-and-time
  |  |  +---w (filterspec)?
  |  |  |  +--:(inline)
  |  |  |  |  +---w (filter-type)?
  |  |  |  |     +--:(subtree)
  |  |  |  |     |  +---w subtree-filter
  |  |  |  |     +--:(xpath)
  |  |  |  |     |  +---w xpath-filter?              yang:xpath1.0
  |  |  |  |     +--:(rfc5277)
  |  |  |  |        +---w filter
  |  |  |  +--:(by-reference)
  |  |  |     +---w filter-ref?                filter-ref
  |  |  +---w (update-trigger)?
  |  |  |  +--:(periodic)
  |  |  |  |  +---w period                     yang:timeticks
  |  |  |  +--:(on-change) {on-change}?
  |  |  |     +---w no-synch-on-start?         empty
  |  |  |     +---w dampening-period           yang:timeticks
  |  |  |     +---w excluded-change*           change-type
  |  |  +---w dscp?               inet:dscp {configured-subscriptions}?
  |  |  +---w subscription-priority?     uint8
  |  |  +---w subscription-dependency?   string
```

- Input parameters mirror those of static subscription, except for receiver specification
- Same underlying groupings

Note: items in blue are new or modified since -00

Model overview: RPCs (output)

```
+---x establish-subscription
|  +--ro output
|     +--ro subscription-result        subscription-result
|     +--ro (result)?
|        +--:(success)
|        |  +--ro subscription-id            subscription-id
|        +--:(no-success)
|           +--ro stream?                    update-stream
|           +--ro encoding?                  encoding
|           +--ro subscription-start-time?   yang:date-and-time
|           +--ro subscription-stop-time?    yang:date-and-time
|           +--ro (filterspec)?
|           |  +--:(inline)
|           |  |  +--ro (filter-type)?
|           |  |     +--:(subtree)
|           |  |     |  +--ro subtree-filter
|           |  |     +--:(xpath)
|           |  |     |  +--ro xpath-filter?           yang:xpath1.0
|           |  |     +--:(rfc5277)
|           |  |        +--ro filter
|           |  +--:(by-reference)
|           |     +--ro filter-ref?               filter-ref
|           +--ro (update-trigger)?
|           |  +--:(periodic)
|           |  |  +--ro period                    yang:timeticks
|           |  +--:(on-change) {on-change}?
|           |     +--ro no-synch-on-start?        empty
|           |     +--ro dampening-period          yang:timeticks
|           |     +--ro excluded-change*          change-type
|           +--ro dscp?         inet:dscp {configured-subscriptions}?
|           +--ro subscription-priority?     uint8
|           +--ro subscription-dependency?   string
```

- “No success” returns parameter settings that should be changed to lead to future success (subscription negotiation)

Note: items in blue are new or modified since -00

Model overview (contd.)

- Not depicted:
  - RPCs for modify, delete subscription
  - Notifications
    - Push updates
      - Push-update, push-change-update
    - Control notifications
      - Subscription started, stopped, suspended, resumed