

IETF YANG Push Lite

draft-wilton-netconf-yang-push-lite-latest

NETCONF WG Interim, Feb 2025

Rob Wilton (presenting)

Including significant input from:

**Thomas (Swisscom), Dan (Bell Canada), Holger (DT)
James (Nokia), Ebben (Juniper), Alex (Insa-Lyon)**



What?

Goal: to build a new YANG Push

MVP = Minimal Viable Push



What Not?

No starting from scratch

(for the protocol ;-)

No reinventing the wheel



Why?

What about RFCs 8639 & 8641?

What about gNMI?



Again, What? E.g.,

- Remove complex parts:
 - Dampening, event frequency negotiation, QoS (except DSCP bits)
- Common encoding for periodic & on-change
- Split up subscriptions/replies
- Simplified filtering?
- Alignment with gNMI, where appropriate
- Simplified NACM conformance?
- Agreed base encoding, filtering, etc?





YP Lite Starting Draft

Github Repo: <https://github.com/rgwilton/draft-yp-observability>

<https://rgwilton.github.io/draft-yp-observability/draft-wilton-netconf-yang-push-lite.html>

Rob started a ***new self-contained draft*** to define YP Lite:

- Copied all relevant text from RFC 8639 and RFC 8641
- Restructured & simplified section orders and text
- Reduced, simplified, & rewrote lots of text and explanations
- The working draft is approx 100 pages, but if reviewing please focus on particular section (for now)
- YP Lite uses **new module namespace for config, notifications and RPCs**

YP Lite Doc Structure

Please focus on

3. Summary of Open Issues

5. Introduction

5.4 Functional changes between YANG Push Lite and YANG Push

5.4.1. **Changed** Functionality

5.4.2. **Removed** Functionality

5.4.3. **Added** Functionality

15. YANG

15.1. **ietf-yp-lite YANG tree**

15.2. ietf-yp-lite YANG Model

Warning: Examples in appendix still out of date

Main structure (still rough)

6. YANG Push Lite **Overview**

7. Definitions

8. **Sensor paths and selection filters**

9. Datastore **Event Streams**

10. **Receivers, Transports, and Encodings**

11. **Setting up and Managing Subscriptions**

12. Subscription **Lifecycle Notifications**

13. Performance, Reliability, and **Subscription Monitoring**

14. Conformance and **Capabilities**

15. **YANG**

15.1. ietf-yp-lite YANG tree

15.2. ietf-yp-lite YANG **Model**

Sensor Paths and Selection Filters

Changed

1. Data model is simplified:
 - I.e., considering removal and additions.
2. Device can internally decompose a configured selection filter into simpler, more specific, filters:
 - Main change for the client is that periodic updates may come in separate messages.
 - Clients would need to listen to end-of-collection notifications to infer objects that have been deleted (or could subscribe to on-change)

Removed

1. Non datastore event streams
2. Filters applied to event stream rather than datastore
3. Proposal: Remove Xpath filtering

Added

1. New basic path filtering:
 - Path with (i) exact value, (ii) regex, or (iii)* for each key
 - I.e., much closer to the actual 'xpath' that is supported today.
2. Multiple target paths in selection filter
3. Proposal: Module schema version selection

Datastore event stream

Changed

1. Only datastore event stream
2. All notifications use draft-xxx-notif-envelope
 - *Also require hostname & sequence number?*
3. New combined *update* notification:
 - Covers both on-change and periodic
 - Structurally closer to periodic than on-change
 - Can have multiple updates (e.g., for on-change) in simple message (shared timestamp)
 - Each update is for an identified subtree (or list entry)
 - Proposed: Can specify a common prefix
4. Target path:
 - JSON instance identifier path
5. On-change is either “*exists with value*” or *deleted*:
 - Delete is replacing an existing entry with empty “{}”
 - *TBD – User-ordered lists.*

Removed

1. On-change patch format
2. Support for on-change events other than “exists with value” or “doesn’t exist”
 - No support for merge/replace/etc.
3. No client configurable dampening
4. Replaying buffered events

Added

1. Common *path prefix* in *update* message
2. Combined periodic + on-change subscription
3. Subscriptions may be decomposed into separate update messages
4. Publisher MAY dampen on-change events

Receivers, Transports, Encodings

Changed

1. Receiver config is separate from subscription:
 - Transport & Encoding are configured under the receiver
2. DSCP is set under receiver
 - *Proposal: Allow DSCP under subscription to override receiver DSCP?*
3. If a subscription uses multiple receivers:
 - MUST all use the same Encoding
 - *Possibly: "MUST all use the same transport/DSCP"?*
 - *Goal: All messages are sent to all receivers with the same sequence-number (except for new receiver-removed message).
I.e., this is a change to the YP subscription management FSM*
4. *TODO – Check what (if any) receiver config should remain under a subscription*

Removed

1. QoS weighting and dependencies between subscriptions:
 - Too complex, not needed.

Added

Setting up and Managing Subscriptions

Changed

1. Receiver config is separate from subscription:
 - Transport & Encoding are configured under the receiver
2. Removed parts of the RFC 8639 data model that are not needed for YP Lite
 - New module namespace
 - Simplification where possible
3. Align configured and dynamic subscription lifecycle
 - E.g., dynamic subscriptions get subscription-started notification
 - No suspend/resume of subscriptions
4. *Proposal: strings for subscription-id (rather than int)*

Removed

1. Dynamic: No modify subscription RPC
 - Can just delete and re-establish the subscription
 - Possible alternative: allow establish-subscription on an existing subscription.
2. Stop-time and replaying buffered events
3. Minimal on-change notifications:
 - I.e., only notify the values that have changed.

Added

1. Combined on-change and periodic subscription
 - *But is this really the same on-change behavior as defined in YP?*

Lifecycle Notifications

Changed

1. More lifecycle alignment between configured and dynamic subscriptions
2. Subscription-started notification is sent to all receivers:
 - So, all receivers can receive the same messages with the same sequence number
 - To better handle receiver redundancy
 - Proposed: Doesn't copy a referenced filter.
3. Subscription-terminated/Subscription-started may be sent in more error cases:
 - E.g., if transport session goes down, or back pressured then subscription will be terminated and restarted (potentially with a replay, if required)

Removed

1. Ability to suspend and resume dynamic subscriptions:
 - Instead, subscription is terminated and must be re-established by the receiver.

Added

1. Proposal: New receiver-removed notification
 - Separate receiver removed notification that is sent to a single receiver when that receiver is removed from a multi-receiver subscription
 - Break in subscription number space won't matter (since it will be last message sent to the receiver)
 - Aim is to differentiate from subscription-terminated
2. Proposal: collection complete notification:
 - Separate message sent after each periodic collection
 - Allows for receivers to detect implicit deletes
 - Opt-in, not sent if not needed.

Subscription Negotiation & Monitoring

Changed

Removed

1. Negotiation and hints for failed dynamic subscriptions:
 - Publishers should refuse new subscriptions that they cannot honor.
 - Publishers will terminate subscriptions they can no longer honor (e.g., a subscription may be fine for 10 interfaces, but not 10k interfaces)
 - Can use operational data model to see overall telemetry load.
 - Expect operators to test/know realistic subscription load & parameters.

Added

1. Proposal: More operational data for subscriptions:
 - E.g., some load/performance figures
 - Perhaps an estimate of total load on telemetry subsystems
 - Aim is to replace complex/fragile subscription negotiation

Capabilities & Conformance

Changed

Conformance:

1. Aiming for stronger interoperability:
 - Both configured and dynamic are optional to implement
 - SHOULD support JSON & CBOR
 - MUST support basic filter selection paths
 - MUST support periodic
 - Any mandatory transport?

Capabilities:

1. YP-Lite to mandate populating operational capabilities
 - New section in draft will describe purpose and high-level structures
 - YP Lite YANG model to augment RFC 9196 for YP-Lite capabilities structure

Removed

1. Proposed: Remove some YANG feature statements
 - Need to decide whether to keep both capabilities and feature statements.
 - Keeping some feature statements probably still makes sense

Added

1. Defines operational capabilities:
 - Still TBD in the draft, but expected to copy existing capabilities, but in a new YP Lite namespace.
2. Advertising optimized subscription paths (tbd)
 - Need to figure out on-change

YP Lite Config/Oper Tree

```

module: ietf-yp-lite
  +--rw datastore-telemetry
  +--rw filters
  |   +--rw filter* [name]
  |   |   +--rw name          string
  |   |   +--rw (filter-spec)?
  |   |   |   +--:(paths)
  |   |   |   |   +--rw paths*   string
  |   |   |   +--:(subtree)
  |   |   |   |   +--rw subtree? <anydata> {ypl:subtree}?
  |   |   |   +--:(xpath)
  |   |   |   |   +--rw xpaths*  yang:xpath1.0 {ypl:xpath}?
  |   |   +--rw subscriptions
  |   |   |   ...
  |   +--rw receivers
  |   |   +--rw receiver* [name]
  |   |   |   +--rw name          string
  |   |   |   +--rw encoding?     encoding
  |   |   |   +--rw dscp?         inet:dscp {dscp}?
  |   |   |   +--rw (notification-message-origin)? {configured}?
  |   |   |   |   +--:(interface-originated)
  |   |   |   |   |   +--rw source-interface?  if:interface-ref
  |   |   |   |   |   |   {interface-designation}?
  |   |   |   |   +--:(address-originated)
  |   |   |   |   |   +--rw source-vrf?       leafref {supports-vrf}?
  |   |   |   |   |   +--rw source-address?   inet:ip-address-no-zone
  |   |   |   +--rw (transport-type)
  
```

```

+--rw subscriptions
  +--rw subscription* [id]
  |   +--rw id          subscription-id
  |   +--rw target
  |   |   +--rw datastore?      identityref
  |   |   +--rw (filter)?
  |   |   |   +--:(by-reference)
  |   |   |   |   +--rw filter-ref  filter-ref
  |   |   |   +--:(within-subscription)
  |   |   |   |   +--rw (filter-spec)?
  |   |   |   |   |   +--:(paths)
  |   |   |   |   |   |   +--rw paths*   string
  |   |   |   |   |   +--:(subtree)
  |   |   |   |   |   |   +--rw subtree? <anydata> {ypl:subtree}?
  |   |   |   |   |   +--:(xpath)
  |   |   |   |   |   |   +--rw xpaths*  yang:xpath1.0 {ypl:xpath}?
  |   |   +--rw dscp?         inet:dscp {dscp}?
  |   |   +--rw purpose?      string
  |   |   |   {configured}?
  |   |   +--ro configured-subscription-state? enumeration
  |   |   |   {configured}?
  |   +--rw receivers
  |   |   +--rw receiver* [name]
  |   |   |   +--rw name          leafref
  |   |   |   +--ro sent-event-records?
  |   |   |   |   yang:zero-based-counter64
  |   |   |   +--ro excluded-event-records?
  |   |   |   |   yang:zero-based-counter64
  |   |   |   +--ro state          enumeration
  |   |   |   +--ro encoding?     encoding
  |   |   |   +--x reset {configured}?
  |   |   |   |   +--ro output
  |   |   |   |   |   +--ro time    yang:date-and-time
  |   +--rw update-trigger
  |   |   +--rw periodic!
  |   |   |   +--rw period          centiseconds
  |   |   |   |   +--rw anchor-time? yang:date-and-time
  |   |   +--rw on-change! {on-change}?
  |   |   |   +--rw sync-on-start?  boolean
  
```

YP Lite RPCs/Notifications

rpcs:

```

+---x establish-subscription
| +---w input
| | +---w target
| | | +---w datastore?          identityref
| | | +---w (filter)?
| | |   +---:(by-reference)
| | |   | +---w filter-ref      filter-ref
| | |   +---:(within-subscription)
| | |   +---w (filter-spec)?
| | |     +---:(paths)
| | |     | +---w paths*       string
| | |     +---:(subtree)
| | |     | +---w subtree?    <anydata> {ypl:subtree}?
| | |     +---:(xpath)
| | |     | +---w xpaths*     yang:xpath1.0 {ypl:xpath}?
| | |     +---w dscp?        inet:dscp {dscp}?
| | |     +---w encoding?    encoding
| | +---ro output
| | +---ro id      subscription-id
|
+---x delete-subscription
| +---w input
| | +---w id      subscription-id
|
+---x kill-subscription
| +---w input
| | +---w id      subscription-id

```

notifications:

```

+---n subscription-started
| +---ro id      subscription-id
| +---ro target
| | +---ro datastore?          identityref
| | +---ro (filter)?
| | | +---:(by-reference)
| | | | +---ro filter-ref      filter-ref
| | | +---:(within-subscription)
| | | +---ro (filter-spec)?
| | |   +---:(paths)
| | |   | +---ro paths*       string
| | |   +---:(subtree)
| | |   | +---ro subtree?    <anydata> {ypl:subtree}?
| | |   +---:(xpath)
| | |   | +---ro xpaths*     yang:xpath1.0 {ypl:xpath}?
| | +---ro dscp?          inet:dscp {dscp}?
| | +---ro purpose?      string {configured}?
|
+---n subscription-terminated
| +---ro id      subscription-id
| +---ro reason   identityref
|
+---n update
| +---ro id?      subscription-id
| +---ro path-prefix?  string
| +---ro snapshot-type? enumeration
| +---ro observation-time? yang:date-and-time
| +---ro updates* [target-path]
| | +---ro target-path  string
| | +---ro data?        <anydata>
| +---ro incomplete?  empty
|
+---n update-completed
| +---ro id      subscription-id

```

Sounds like a lot of changes ...

1. The core part of YP Lite is the same as YP
2. Merges in various extension drafts
3. Same basic data model (but in a new namespace, cleaned up and simplified)
4. Modifications to periodic notifications are minor, *except* splitting up updates to separate messages
5. Modifications to on-change are perhaps *a bit more significant* (depending on where we end up)
6. Same core subscription FSM (but less states, and more alignment between config and dynamic)



Next steps

Proposal:

- Getting interested authors together:
 - To discuss and resolve issues
- Goal: Move reasonably fast
 - Rob willing to invest time to edit draft

