

An Architecture for YANG-Push to Message Broker **Integration**

draft-ietf-nmop-yang-message-broker-integration-01

Motivation and architecture of a native
YANG-Push notifications and YANG Schema integration
into Message Broker and YANG Schema Registry

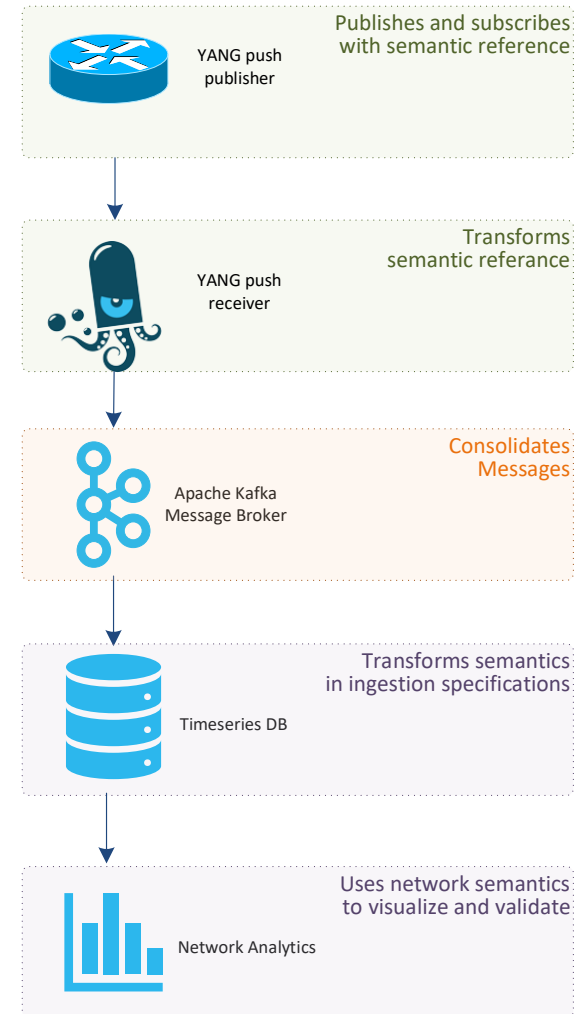
thomas.graf@swisscom.com
ahmed.elhassany@swisscom.com

9. June 2024

From YANG-Push to Network Analytics

Aiming for an automated processing pipeline

- **A network operator aims for:**
 - An **automated data processing pipeline** which starts with YANG-Push, consolidates at Data Mesh and ends at Network Analytics.
 - Operational metrics where **IETF defines the semantics.**
 - Analytical metrics where **network operators gain actionable insights.**
- **We achieve this by integrating YANG-Push into Data Mesh to:**
 - Produce metrics from networks **with timestamps when network events were observed.**
 - Hostname, publisher ID and sequence numbers help us to understand **from where metrics were exported and measure its delay and loss.**
 - Forward **metrics unchanged** from networks
 - **Learn semantics** from networks and validate messages.
 - **Control semantic** changes end to end.



An Architecture for YANG-Push to Apache Kafka Integration

Status, Summary and Next steps

Status

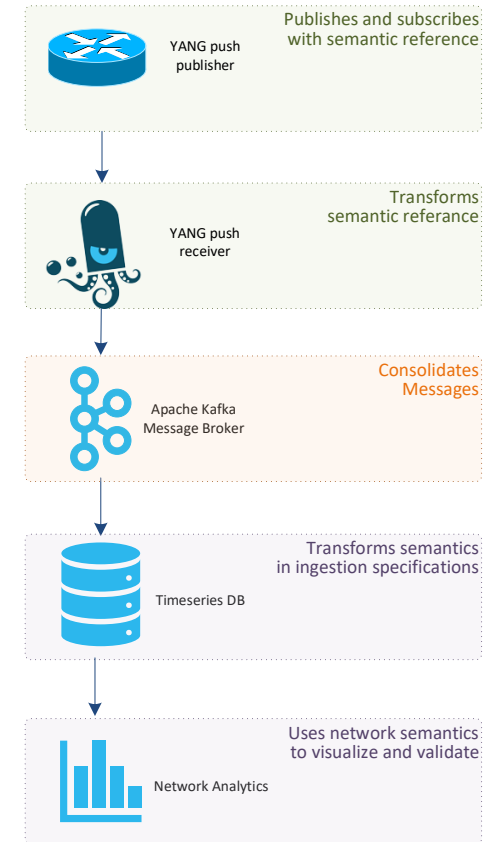
- Document adopted at NMOP.
- Feedback from Dhruv, Andy, Fend and Qin addressed in -01. Thank you very much!

Changes in -01

- Expanded last paragraph in introduction section to detail manual work currently needed in the end-to-end data processing chain due to missing YANG schema
- Figure 1 in Section 3 and Section 3.1 now considers in step 1 to perform the YANG-Push notification capabilities described in Section 3 of RFC 9196
- Added the message broker component in figure 1 in Section 3
- Added section 3.8 describing that observation-time is used for times series metric indexing
- Moved Section 4 and 5 to appendix
- Used the boiler plate from RFC 7942 and moved section before Security Considerations
- Applied RFC 8792 to handle long lines
- Section 3.1 describes that in the described architecture both, dynamic and configured YANG-Push subscriptions are supported. To add clarity, an additional paragraph was added detailing on how being subscribed, messages are published in same or different transport session.

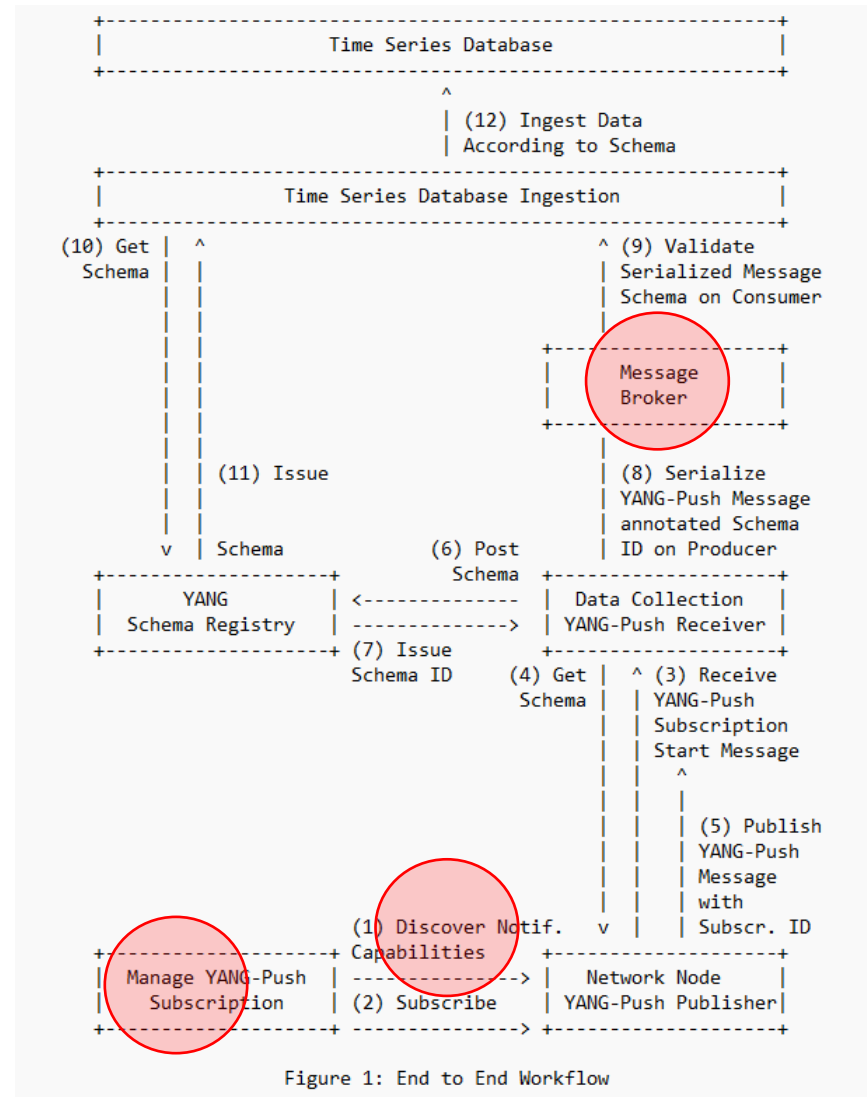
Next Steps

- **Looking forward for review and comments.**



Elements of the Architecture

Workflow Diagram



- **Network Orchestration** subscribes to YANG datastore.
- **Network Node** informs Data Collection on subscription state and publishes YANG metrics with YANG-Push.
- **Data Collection** obtains for each subscription the YANG module dependencies and the YANG modules on the network node, registers it in the YANG Schema Registry and prefixes the forwarded YANG notifications with the obtained schema ID.
- **YANG Schema Registry** issues for a Message Broker subject a schema ID for each new schema tree, compares a new schema tree with an existing and versions it.
- **Time Series Database Ingestion** consumes YANG-Push notifications from Message Broker, obtains schema tree from YANG schema registry, validates YANG notifications against schema and uses schema to populate into database table.

Netconf Notifications

Define YANG module

```
module: ietf-notification

    structure notification:
        +-- eventTime      yang:date-and-time

<notification
xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2023-02-04T16:30:11.22Z</eventTime>
  <push-update xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push">
    <id>1011</id>
    <datastore-contents>
      <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-
interfaces">
        <interface>
          <name>eth0</name>
          <oper-status>up</oper-status>
        </interface>
      </interfaces>
    </datastore-contents>
  </push-update>
</notification>
```

- **YANG model for NETCONF Event Notifications**, [draft-ahuang-netconf-notif-yang](#), updates RFC 5277 by defining the schema as a YANG module.
- This enables YANG-push to define semantics for the entire YANG push message and use other encodings than XML such as YANG-JSON RFC 7951 or YANG-CBOR RFC 9264.
- **Changes in -05:** Updates and describes relationship to RFC 5277, RFC 8639, RFC 7951 and RFC 9254 in terms of notification structure.

Subscription State Change Notifications

RFC 8641 Extensions

```
{
  "ietf-notification:notification": {
    "eventTime": "2023-03-25T08:30:11.22Z",
    "ietf-notification-sequencing:sysName": "example-router",
    "ietf-notification-sequencing:sequenceNumber": 1,
    "ietf-subscribed-notification:subscription-started": {
      "id": 6666,
      "ietf-yang-push:datastore": "ietf-datastores:operational",
      "ietf-yang-push:datastore-xpath-filter": "/if:interfaces",
      "ietf-yang-push-revision:revision": "2014-05-08",
      "ietf-yang-push-revision:module-name": "ietf-interfaces",
      "ietf-yang-push-revision:revision-label": "",
      "ietf-distributed-notif:message-observation-domain-id": [1,2],
      "transport": "ietf-udp-notif-transport:udp-notif",
      "encoding": "encode-json",
      "ietf-yang-push:periodic": {
        "ietf-yang-push:period": 100
      }
    }
  }
}
```

Figure 3: JSON YANG-Push Example for a subscription-started notification message

- **Support of Versioning in YANG Notifications**
Subscription, [draft-ietf-netconf-yang-notifications-versioning](#), adds the ability to subscribe to a specific revision or latest-compatible-semversion. Extends the YANG-Push Subscription State Change Notifications so that the receiver learns on top of xpath and the sub-tree filter also the YANG module name, revision and revision-label.
- **Changes in -05:** Changed ietf-yang-push.yang augmentation to resolve YANG issue that within a "case" statement identifiers need to be unique.
- **Support of Hostname and Sequencing in YANG Notifications**, [draft-tgraf-netconf-notif-sequencing](#), extends the NETCONF notification defined in RFC5277 with sysName, publisherId and sequenceNumber.
- **Changes in -05:** Defined new NETCONF and YANG-Push notification capabilities and described how a systems discovers them.

Push-Update and Push-Change-Update Notifications

RFC 8641 Extensions

```
{
  "ietf-notification:notification": {
    "eventTime": "2023-03-25T08:30:11.22Z",
    "ietf-notification-sequencing:sysName": "example-router",
    "ietf-notification-sequencing:sequenceNumber": 1,
    "ietf-yang-push:push-update": {
      "id": 6666,
      "ietf-yp-observation-time:observation-time": "2023-02-04T16:30:09.44Z",
      "ietf-yp-observation-time:point-in-time": "current-accounting",
      "datastore-contents": {
        "ietf-interfaces:interfaces": [
          {
            "interface": {
              "name": "eth0",
              "type": "iana-if-type:ethernetCsmacd",
              "oper-status": "up",
              "mtu": 1500
            }
          }
        ]
      }
    }
  }
}
```

Figure 4: JSON YANG-Push Example for a push-update notification message

- **Support of Hostname and Sequencing in YANG Notifications**, [draft-tgraf-netconf-notif-sequencing](#), , extends the NETCONF notification defined in RFC5277 with sysName, publisherId and sequenceNumber.
- **Changes in -05:** Defined new NETCONF and YANG-Push notification capabilities and described how a systems discovers them.
- **Support of Network Observation Timestamping in YANG Notifications**, [draft-tgraf-netconf-yang-push-observation-time](#), extends YANG-Push push-update notifications with observation-time and state-changed-observation-time.
- **Changes in -01:** Changed semantics; observation-time describes when and point-in-time at which point in time. Added new YANG-Push notification capabilities.

Augmented-by Addition

YANG Library Extension

```
module: ietf-yang-library
  +--ro yang-library
  |   +--ro module-set* [name]
  |   |   +--ro name string
  |   |   +--ro module* [name]
  |   |   |   +--ro name yang:yang-
  identifier
  |   |   |   +--ro revision? revision-
  identifier
  |   |   |   +--ro namespace inet:uri
  |   |   |   +--ro location* inet:uri
  |   |   |   +--ro submodule* [name]
  |   |   |   |   +--ro name yang:yang-identifier
  |   |   |   |   +--ro revision? revision-identifier
  |   |   |   |   +--ro location* inet:uri
  |   |   |   +--ro feature* yang:yang-
  identifier
  |   |   |   +--ro deviation* ->
  ../../module/name
  |   |   |   +--ro yanglib-aug:augmented-by*
  |   |   |   ->
  ../../yanglib:module/name
```

- **Augmented-by Addition into the IETF-YANG-Library,** [draft-lincla-netconf-yang-library-augmentation](#), enables that augmented-by YANG modules can now be discovered in YANG Library [RFC 8525](#).

Validate anydata schema subtree with YANG Library

RFC 7950 Extension

notifications:

```
+---n push-update
|   +--ro id?                sn:subscription-id
|   +--ro datastore-contents? <anydata>

{
  "ietf-yang-push:push-update": {
    "id": 89,
    "datastore-contents": {
      "ietf-interfaces:interfaces": {
        "interface": [
          {
            "name": "eth0",
            "oper-status": "down"
          }
        ]
      }
    }
  }
}
```

- **Validating anydata in YANG Library context,** [draft-aelhassany-anydata-validation](#), enables that anydata modeled nodes can be validated with YANG Library [RFC 8525](#).

[RFC 7950](#)

7.10. The "anydata" Statement

The "anydata" statement defines an interior node in the schema tree. It takes one argument, which is an identifier, followed by a block of substatements that holds detailed anydata information.

The "anydata" statement is used to represent an unknown set of nodes that can be modeled with YANG, except anyxml, but for which the data model is not known at module design time. **It is possible, though not required, for the data model for anydata content to become known through protocol signaling or other means that are outside the scope of this document.**

Open Points from IETF 119

Addressed at IETF 120

Open Point 1: datastore-contents in push-update or the value in push-change-update uses anydata as data type which contents does not have a schema defined. [draft-aelhassany-anydata-validation](#) addresses that anydata modeled nodes can be validated with YANG Library [RFC 8525](#).

Open Point 2: Definitions how NOTIFICATIONS are encoded in NETCONF are defined in Section 4.2.10 of [RFC 7950](#). However, specifications for encoding in JSON and CBOR are missing [RFC 7951](#). Confirm finding and propose how this needs to be addressed.

Open Point 3: Test with running code wherever with [draft-ietf-netconf-yang-notifications-versioning](#) and [draft-lincla-netconf-yang-library-augmentation](#) all datastore-subtree-filter or datastore-xpath-filter referenced YANG modules and their dependencies can be fully identified.

Milestones

IETF 115 - 120

IETF 115: Official Project Kickoff. Introduced [draft-ietf-netconf-yang-notifications-versioning](#).

IETF 116: YANG module with augmentations can be registered in Confluent Schema Registry with YANG extension. [draft-tgraf-netconf-notif-sequencing](#), [draft-tgraf-netconf-yang-push-observation-time](#) and [draft-ahuang-netconf-notif-yang](#) introduced.

IETF 118: All relevant YANG modules for a subscribed xpath can be determined through the YANG Library [RFC 8525](#) and retrieved through NETCONF <get-schema> rpc calls according to [RFC 6022](#). Gap in YANG library addressed in [draft-lincl-netconf-yang-library-augmentation](#).

IETF 119: [draft-aelhassany-anydata-validation](#) addresses that anydata modeled nodes can be validated with YANG Library [RFC 8525](#). 6WIND VSR and Huawei VRP YANG-Push and open-source [draft-lincl-netconf-yang-library-augmentation](#) implementation validated at hackathon.

IETF 120: 6WIND VSR, Huawei VRP and Cisco IOS XR YANG-Push publisher and [draft-aelhassany-anydata-validation](#) implementation validated at hackathon. Running code proofed that with [draft-ietf-netconf-yang-notifications-versioning](#) and [draft-lincl-netconf-yang-library-augmentation](#) all datastore-subtree-filter or datastore-xpath-filter referenced YANG modules and their dependencies can be fully identified.

YANG-Push Implementation Status

IETF 120

	6WIND VSR	Huawei VRP	Cisco IOS XR
RFC 8641 YANG-Push	x	x	x
draft-ietf-netconf-udp-notif	x	x	
draft-ietf-netconf-distributed-notif	x	x	
draft-ietf-netconf-yang-notifications-versioning	x	x	
draft-tgraf-netconf-notif-sequencing	x		
draft-tgraf-netconf-yang-push-observation-time	x		
RFC 7895 YANG Module Library		x	
RFC 8525 YANG Library	x		x
draft-lincla-netconf-yang-library-augmentation			



Industry Colaboration

On YANG Push to Apache Kafka integration

