

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

draft-ietf-nmop-yang-message-broker-integration-05

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
alex.huang-feng@insa-lyon.fr

25. October 2024

Agenda Items

- Document status and next steps
- Implementation status
- Related YANG-Push integration and specification gaps status

An Architecture for YANG-Push to Message Broker Integration

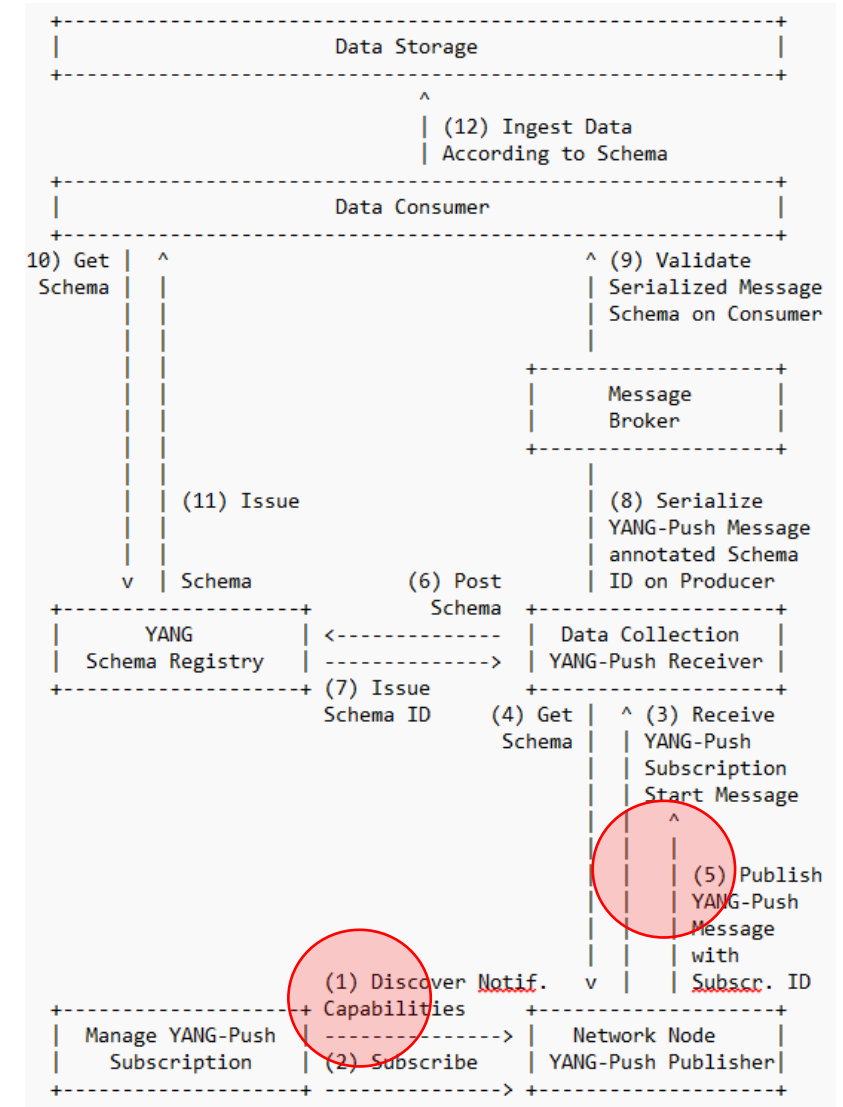
Status, Summary and Next steps

Changes in -05

- Section 4.1 covers now
 - [draft-netana-netconf-yp-transport-capabilities](#) which helps the NETCONF, RESTCONF client performing YANG-Push subscription to discover the YANG-Push transport, encoding and security capabilities of the YANG-Push publisher.
 - [draft-netana-netconf-notif-envelope](#) defines a new extensible notification structure, defined in YANG, for use in YANG-Push Notification messages enabling any YANG compatible encodings such as XML, JSON or CBOR. Replaces and consolidates draft-ahuang-netconf-notif-yang and draft-tgraf-netconf-notif-sequencing based on the outcome of the last NETCONF interim.
- Milestones for IETF 121 and 122 have been added.
- Open Points and YANG-Push notification examples have been updated reflecting the changes from [draft-netana-netconf-yp-transport-capabilities](#) and [draft-netana-netconf-notif-envelope](#).

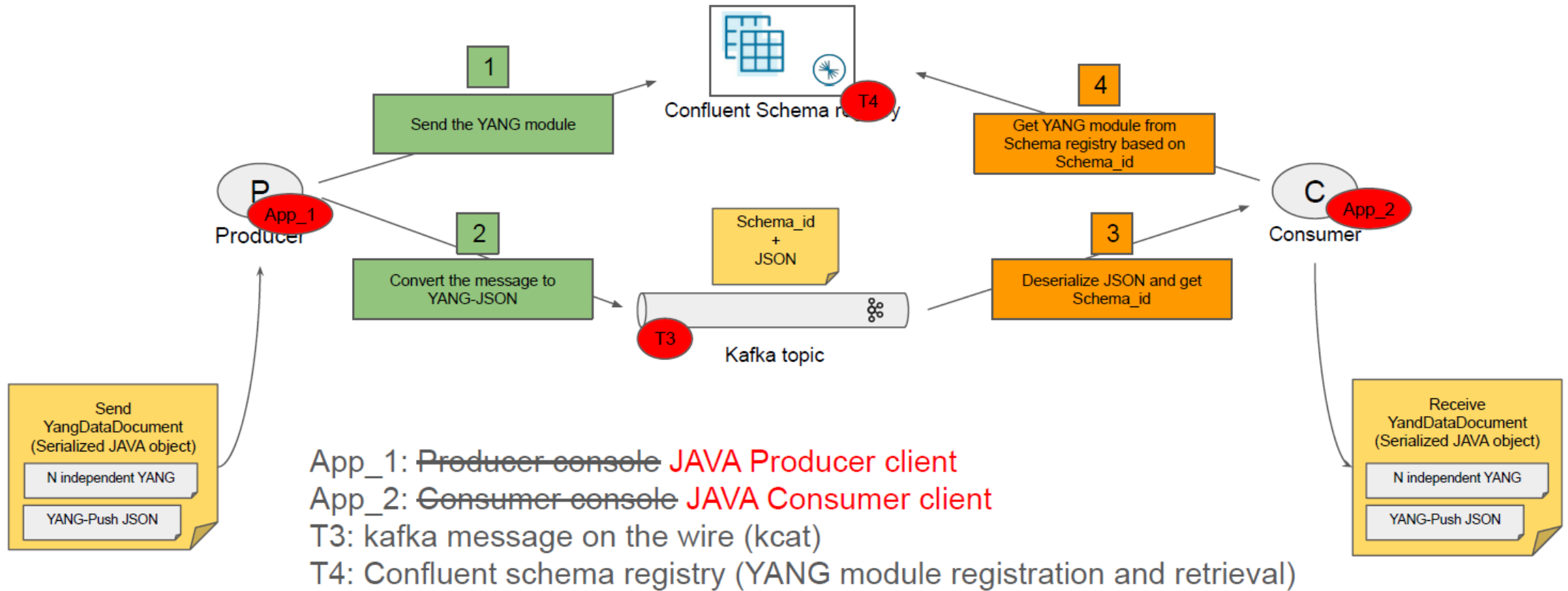
Next Steps

- Looking forward for review and comments.



An Architecture for YANG-Push to Message Broker Integration

Implementation Status



Address YANG Specification and Integration Gaps

Aiming for an automated data processing pipeline

YANG Specifications Gaps:

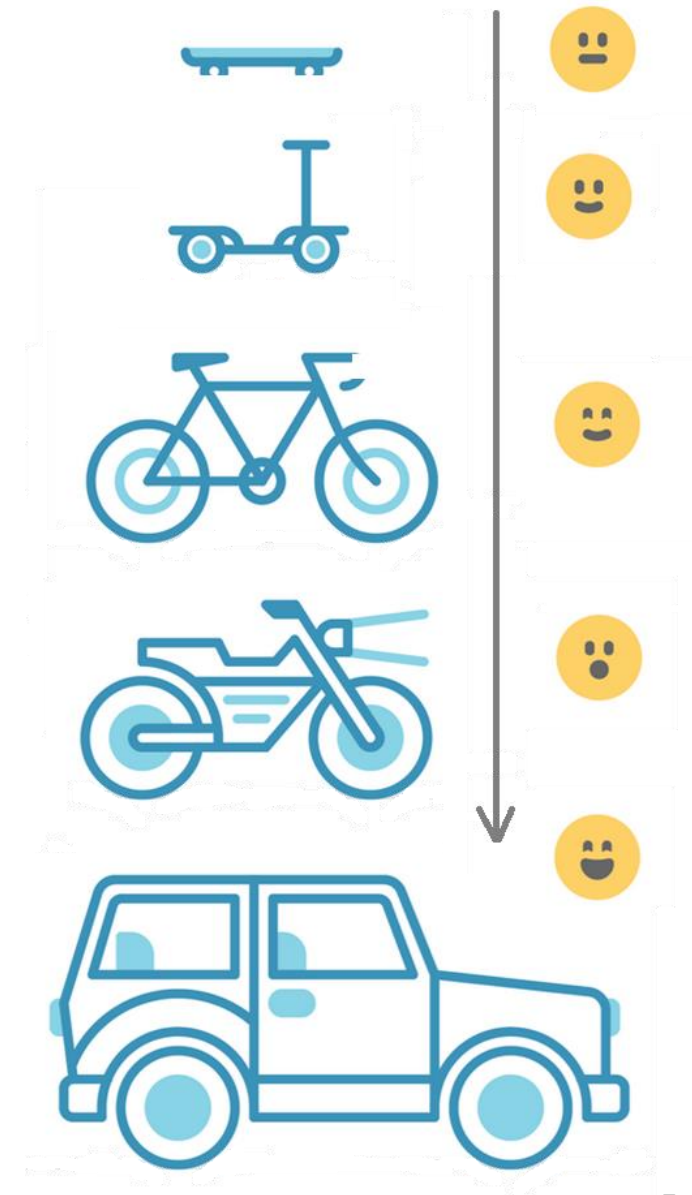
- Extensible YANG model for YANG-Push Notifications
[draft-netana-netconf-notif-envelope](#)
- YANG Notification Transport Capabilities
[draft-netana-netconf-yp-transport-capabilities](#)
- Validating anydata in YANG Library context
[draft-aelhassany-anydata-validation](#)

YANG Integration Gaps:

- Support of Network Observation Timestamping in YANG Notifications
[draft-tgraf-netconf-yang-push-observation-time](#)
- Support of Versioning in YANG Notifications Subscription
[draft-ietf-netconf-yang-notifications-versioning](#)
- Augmented-by Addition into the IETF-YANG-Library
[draft-ietf-netconf-yang-library-augmentation](#)

YANG Simplification:

- YANG-Push Operational Data Observability Enhancements
[draft-wilton-netconf-yp-observability](#)



YANG Notification **Transport Capabilities**

Extending System Capabilities for YANG-Push Configured Subscription Transport

```
module: ietf-notification-transport-capabilities

augment /sysc:system-capabilities/notc:subscription-capabilities:
  +--ro transport-capabilities
    +--ro transport-capability* [transport-protocol]
      +--ro transport-protocol identityref
      +--ro security-protocol? identityref
      +--ro encoding-format* identityref

augment "/sysc:system-capabilities/notc:subscription-capabilities" {
  description "Add system level capability.";
  container transport-capabilities {
    description "Capabilities related to YANG-Push transports.";
    list transport-capability {
      key "transport-protocol";
      description "Capability list related to notification transport capabilities.";
      leaf transport-protocol {
        type identityref {
          base sn:transport;
        }
        description "Supported transport protocol for YANG-Push.";
      }
      leaf security-protocol {
        type identityref {
          base security-protocol;
        }
        description "Type of secure transport.";
      }
      leaf-list encoding-format {
        type identityref {
          base sn:encoding;
        }
        description "Supported encoding formats.";
      }
    }
  }
}
```

- [draft-netana-netconf-yp-transport-capabilities](#) augments System Capabilities model and provides additional transport related attributes associated with system capabilities:
- Specification of transport protocols the client can request to establish a [draft-ietf-netconf-udp-notif](#) or [draft-ietf-netconf-https-notif](#) configured transport connection;
- Specification of transport encoding, such as JSON or XML as defined in [RFC 8040](#) or CBOR as defined in [RFC 9254](#) the client can request to encode YANG notifications;
- Specification of secure transport mechanisms that are needed by the client to communicate with the server such as DTLS as defined in [RFC 9147](#) TLS as defined in [RFC 8446](#) or SSH as defined in [RFC 4254](#);

Extensible YANG model for YANG-Push Notifications

For XML, JSON or CBOR encoded messages with hostname and sequence-number

```
notifications:
  +---n envelope
    +--ro event-time                yang:date-and-time
    +--ro hostname?                 inet:host
    |                               {notification-hostname-sequence-number}?
    +--ro sequence-number?          yang:counter32
    |                               {notification-hostname-sequence-number}?
    +--ro notification-contents?    <anydata>
```

```
{
  "ietf-yp-notification-envelope": {
    "event-time": "2023-03-25T08:30:11.22Z",
    "hostname": "example-router",
    "sequence-number": 1,
    "notification-contents": {
      "ietf-yang-push:push-update": {
        "id": 6666,
        "datastore-contents": {
          "ietf-interfaces:interfaces": [
            {
              "interface": {
                "name": "eth0",
                "type": "iana-if-type:ethernetCsmacd",
                "oper-status": "up",
                "mtu": 1500
              }
            }
          ]
        }
      }
    }
  }
}
```

- [draft-netana-netconf-notif-envelope](#) defines new extensible notification structure, defined in YANG, for use in YANG-Push Notification messages enabling any YANG compatible encodings such as XML [RFC 7950](#), JSON [RFC 7951](#) or CBOR [RFC 9264](#).
- New notification envelope can be enabled in "ietf-subscribed-notification" [RFC 8639](#).
- Capability can be discovered through 'ietf-notification-capabilities' [RFC 9196](#).
- Supports the following notification metadata extensions
 - **hostname:** Describes the node's hostname according to the 'sysName' object definition in RFC 1213 from where the message was published from. This value is usually configured on the node by the administrator to uniquely identify the node in the network.
 - **sequence-number:** Generates a unique sequence number for each published message by the publisher process. The number counts up at every published notification message as described in RFC 9187.

YANG-Push Operational Data Observability Enhancements

Simplifies by combining periodic and on-change subscription

```
module: ietf-yp-ext

augment /sn:subscription-started/yp:update-trigger:
  +--:(periodic-and-on-change) {yp:on-change}?
    +-- periodic-and-on-change!
      +-- period                yp:centiseconds
      +-- anchor-time?         yang:date-and-time
      +-- dampening-period?    yp:centiseconds
      +-- sync-on-start?       boolean
      +-- excluded-change*     yp:change-type
augment /sn:subscription-started:
  +--ro common-notification-format?  boolean
augment /sn:subscription-modified/yp:update-trigger:
  +--:(periodic-and-on-change) {yp:on-change}?
    +-- periodic-and-on-change!
      +-- period                yp:centiseconds
      +-- anchor-time?         yang:date-and-time
      +-- dampening-period?    yp:centiseconds
      +-- sync-on-start?       boolean
      +-- excluded-change*     yp:change-type
augment /sn:subscription-modified:
  +--ro common-notification-format?  boolean
augment /sn:subscriptions/sn:subscription/yp:update-trigger:
  +--:(periodic-and-on-change) {yp:on-change}?
    +--rw periodic-and-on-change!
      +--rw period                yp:centiseconds
      +--rw anchor-time?         yang:date-and-time
      +--rw dampening-period?    yp:centiseconds
      +--rw sync-on-start?       boolean
      +--rw excluded-change*     yp:change-type
augment /sn:subscriptions/sn:subscription:
  +--rw common-notification-format?  boolean
```

- To reduce complexities in modelling the operational state, the following two YANG-Push enhancements are proposed:
 - A new YANG-Push encoding format that can be used for both on-change and periodic subscriptions that reports the data from the subscription filter point.
 - A combined periodic and on-change subscription that reports events on a periodical cadence and also if changes to the data have occurred.

```
notifications:
+---n update
  +--ro id?                sn:subscription-id
  +--ro subscription-path? yang:xpath1.0
  +--ro target-path?       string
  +--ro snapshot-type?     enumeration
  +--ro observation-time?  yang:date-and-time
  +--ro datastore-snapshot? <anydata>
  +--ro incomplete?        empty
```

- This removes the YANG Patch format [RFC 8072](#) dependency and eases the message broker integration.
- Allows the YANG-Push publisher to split a subscription into smaller child subscriptions for more efficient independent and concurrent processing. Reuses the ideas from [draft-ietf-netconf-distributed-notif](#). Child subscriptions remain encoded from the same subscription point.



♥ Yang ♥♥♥
KafKa ♥

Handling Operational YANG Modelled Data

State of the Union

Nowadays network operators are using **machine and human readable YANG** [RFC 7950](#) to model their configurations and obtain YANG modelled data from their networks.

Network operators organizing their data in a Data Mesh where a message broker such as Apache Kafka facilitates the exchange of messages among data processing components.

Today, subscribing to a YANG datastore, publishing a YANG modeled notifications message from the network and viewing the data in a time series database, **manual labor is needed to perform data transformation** to make a message broker and its data processing components with YANG notifications interoperable.

« Even though YANG is intend to ease the handling of data, **this promise has not yet been fulfilled** for Network Telemetry [RFC 9232](#) »

From YANG-Push to Network Analytics

Aiming for an automated data 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.

