

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

draft-ietf-nmop-yang-message-broker-integration-06

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

13. February 2025

# Agenda Items

- Subscribing to YANG – State of the Union
- Motivation, Architecture and Implementation Status
- YANG-Push Notifications and Capability Discovery Developments
- Who we are and what MVP's are
- Implementation Status and upcoming IETF 122 Hackathon

# IETF YANG-Push

## A 22 years journey

### IAB Workshop

Defines operators' requirements in RFC 3535 to lifecycle CLI and SNMP. YANG, Netconf and Restconf development started.

2002

### YANG 1.0

Specified in RFC 6020. 1.1 in RFC 7950.

2010

### gNMI

gNMI was presented to IETF NETCONF and implementations started at major network vendors.

2017

2015

### IETF YANG-Push Specification Started

Development of RFC 8639 and RFC 8641 started at IETF NETCONF.

2019

### IETF YANG-Push Specification Finished

Development of RFC 8639 and RFC 8641 concluded at IETF NETCONF without any major network vendor implementations.

### Data Mesh Integration

Vendor-specific implementations and IETF YANG-Push are hard to manage. New requirements emerged for integrating with the message broker and an automated data processing chain. New specifications are proposed to resolve these challenges.

2022

2024

### IETF YANG-Push Major Implementations Started

Questions arise. Proposing a simplified IETF YANG-Push and an Agile Incremental Driven Development.

# 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 organize their data in a Data Mesh** where a message broker such as Apache Kafka facilitates the exchange of messages among data processing components.

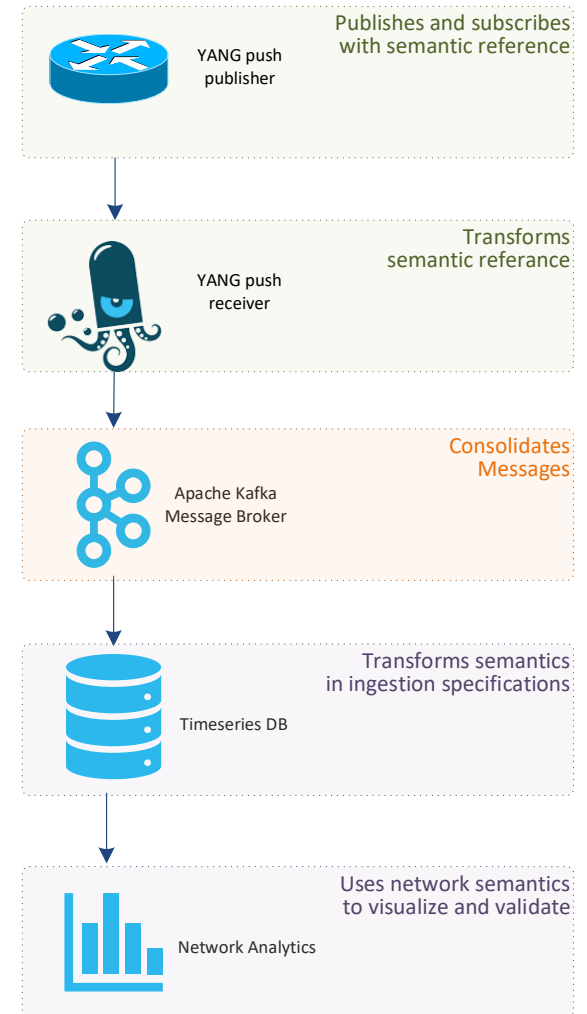
Today, subscribing to a YANG datastore, publishing a YANG modeled notifications message from the network and viewing the data in a time series database, **manual labor is needed to perform data transformation** to make a message broker and its data processing components with YANG notifications interoperable.

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

# From YANG-Push to Network Analytics

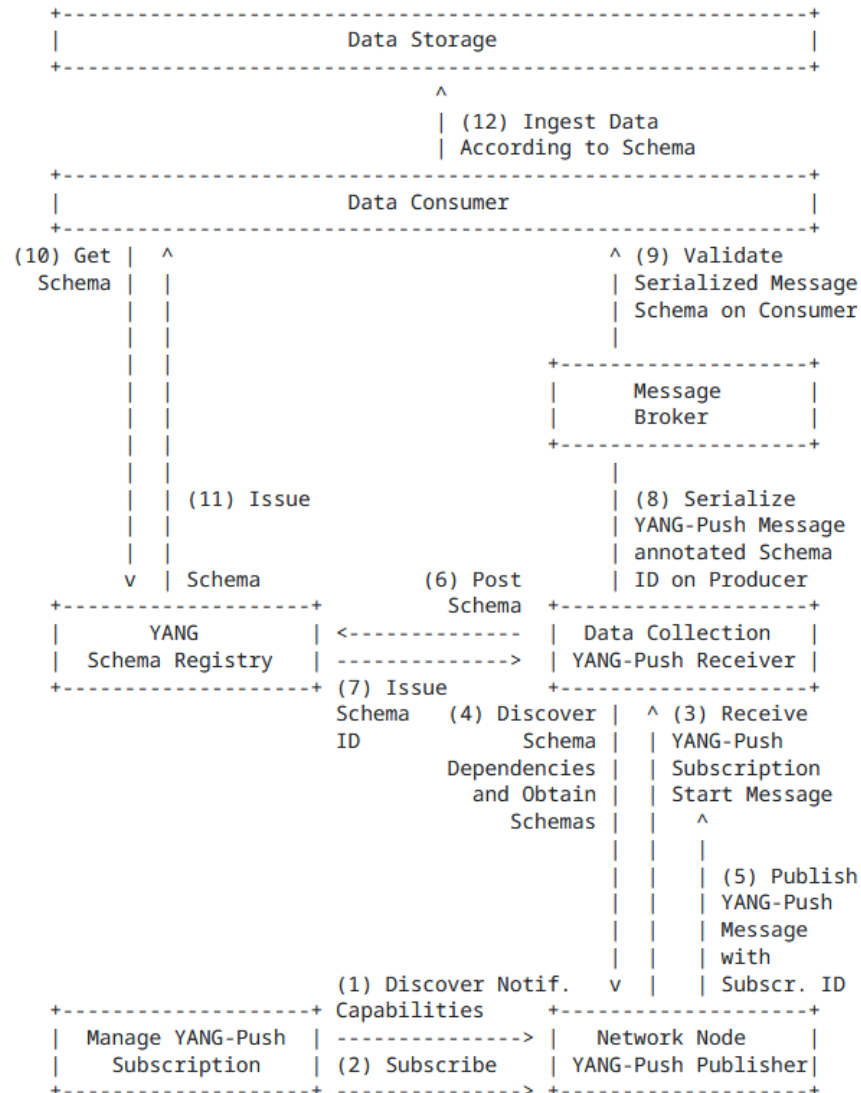
Aiming for an automated data processing pipeline

- **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.



# Elements of the Architecture

## And Document Relationships



### 4.1. YANG-Push Subscription

Thanks to [RFC 9196](#) based YANG modules describing capabilities for systems and notifications, transport ([draft-netana-netconf-yp-transport-capabilities](#)), notification extensions ([draft-netana-netconf-notif-envelope](#)) and supported xpaths can be discovered.

### 4.2. YANG-Push Publisher

For configured subscriptions, [udp-notif](#), [distributed-notif](#) and [https-notif](#) are the [RFC 8639](#), [RFC 8641](#) based YANG-Push transport protocols. [draft-netana-netconf-notif-envelope](#) provides the notification header with hostname, sequence and observation timestamp extensions.

### 4.3. YANG-Push Receiver

Discovers subscription state changes by detecting changes in xpath/sub-tree, module name, revision, revision-label and YANG library content-id defined in [draft-ietf-netconf-yang-notifications-versioning](#) and discovers YANG schema tree through [YANG library](#) with [ietf-netconf-yang-library-augmentedby](#) extension.

### 4.6. YANG Message Broker Consumer

Validates YANG notifications against schema. Anydata described in [Section 7.10 of RFC 7950](#), validation described in [draft-aelhassany-anydata-validation](#).

### 4.7. YANG Data Consumer

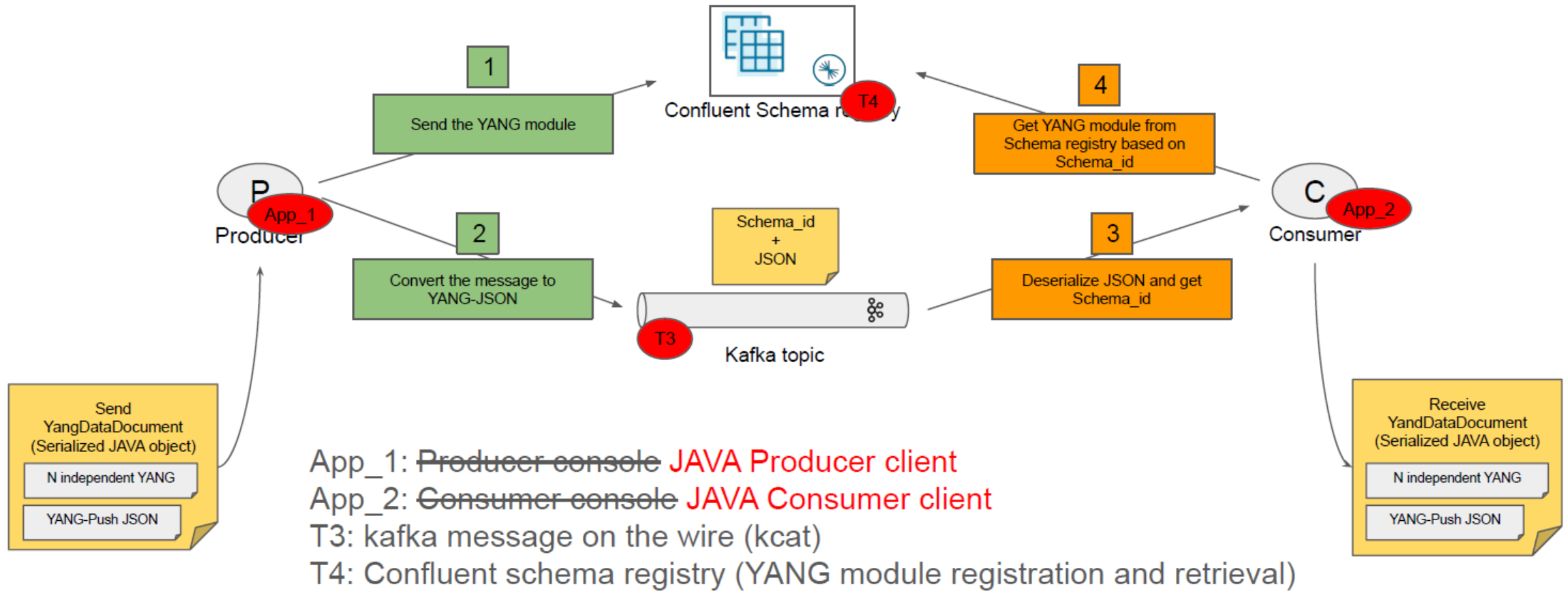
Thanks to hostname, sequence-number and event-time from [draft-netana-netconf-notif-envelope](#), loss and serialization delay can be measured.

### 4.8. YANG Data Storage

Metrics can be indexed in timeseries database thanks to observation timestamp defined in [draft-netana-netconf-notif-envelope](#).

# An Architecture for YANG-Push to Message Broker Integration

IETF 120 Implementation Status



# Extensible YANG model for YANG-Push Notifications

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

```
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>

module: ietf-yp-observation-time
augment /yp:push-update:
+---ro observation-time?        yang:date-and-time
+---ro point-in-time?           enumeration
augment /yp:push-change-update:
+---ro observation-time?        yang:date-and-time
+---ro point-in-time?           enumeration
augment /sysc:system-capabilities/notc:subscription-capabilities:
+---ro yang-push-observation-supported?
    inotifseq:notification-support
    {yang-push-observation-timestamp}?
{
  "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,
        "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
              }
            }
          ]
        }
      }
    }
  }
}
```

- [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.
  - **observation-time:** Describes the measurement observation time for the "push-update" notification in a "periodical" and for the "push-change-update" notification in a "on-change" subscription.
  - **point-in-time:** Describes at which point in time the value of observation-time was observed.



# Support of **Versioning** in YANG Notifications Subscription

For subscription state change notification messages

```
module: ietf-yang-push-revision
augment /sn:establish-subscription/sn:input:
  +---w module-version-config* [module-name]
    +---w module-name          yang:yang-identifier
    +---w revision?            rev:revision-date-or-label
    +---w revision-label?      ysver:version

augment /sn:subscription-started:
  +--ro module-version* [module-name]
  |   {yang-push-revision-supported}?
  | +--ro module-name      yang:yang-identifier
  | +--ro revision         rev:revision-date
  | +--ro revision-label?  ysver:version
  +--ro yang-library-content-id? -> /yanglib:yang-library/content-id
    {yang-push-revision-supported}?

{
  "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-yang-push-revision:yang-library-content-id": "1",
      "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
      }
    }
  }
}
```

- **Network operators need to control semantics in its data processing pipeline. That includes YANG-Push.**
  - This is today only possible during YANG-Push subscription but not when nodes are being upgraded or when messages are being published for configured subscription.
  - [draft-ietf-netconf-yang-notifications-versioning](#) extends the YANG push subscription and publishing mechanism defined in [RFC 8641](#):
    - **By adding the ability to subscribe to a specific revision or latest-compatible-semversion of one or more yang modules.**
    - **By extending the YANG push Subscription State Change Notifications Message** so that the YANG push receiver learns beside the xpath and the sub-tree filter also the yang module name, revision, revision-label and the yang-library-content-id.
- With YANG Library content-id a YANG-Push receiver is now able to detect changes in the YANG library. This includes also the imported YANG modules of the subscribed xpath.
- Extends [RFC 9196](#) defined subscription-capabilities with a **yang-push-module-revision-supported** leaf.

# 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](#);

# 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
```

- With YANG-Push xpath or sub-tree a part of the YANG-Push data tree is subscribed.
- With YANG Library the relationship among the subscribed YANG modules can be determined from the top of the YANG tree. **What is missing is the ability to discover dependencies within the YANG tree.**
- [draft-lincla-netconf-yang-library-augmentation](#) extends the YANG library defined in [RFC 8525](#):
  - **By adding augmented-by YANG module relation.**

# 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"
          }
        ]
      }
    }
  }
}
```

- With YANG-Push xpath or sub-tree a part of the YANG-Push data tree is subscribed. The subscribed YANG datastore content is published as anydata, even though the content has a valid schema.
- RFC 7950 lacks specification how the data model of anydata content is exposed through YANG library defined in [RFC 8525](#).
- [draft-aelhassany-anydata-validation](#) extends [RFC 7950](#) by describing:
  - **How anydata can be validated with YANG Library.**

### [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.

# 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.

# Addressing YANG Specification and Integration Gaps

9 documents at NMOP, NETCONF and NETMOD

## YANG-Push Transport Gaps:

- UDP-based Transport for Configured Subscriptions  
[draft-ietf-netconf-udp-notif](#)
- Subscription to Distributed Notifications  
[draft-ietf-netconf-distributed-notif](#)

## YANG-Push 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-Push Integration Gaps and Arch:

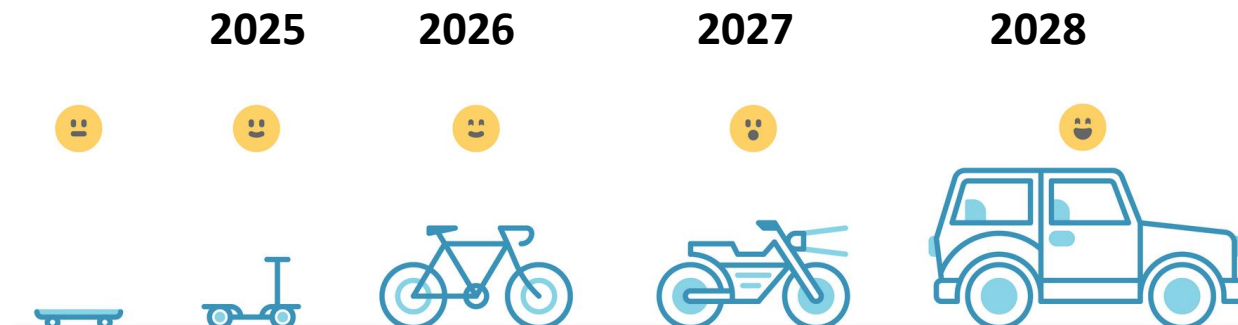
- Support of Versioning in YANG Notifications Subscription  
[draft-ietf-netconf-yang-notifications-versioning](#)
- Augmented-by Addition into the IETF-YANG-Library  
[draft-ietf-netconf-yang-library-augmentation](#)

## YANG-Push Simplification:

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

## YANG-Push Message Broker:

- An Architecture for YANG-Push to Message Broker Integration  
[draft-netana-nmop-yang-message-broker-integration](#)



# IETF YANG-Push Implementations and Next Steps

## Who we are and what we like...

The group consisting of: **34 colleagues** from Bell Canada, Deutsche Telekom, NTT International, Swisscom, Huawei, Cisco, 6Wind, Ciena Blueplanet, Juniper, Nokia, and INSA Lyon.

Development on first major vendor implementations started at IETF 118. Throughout IETF 119 - 121, vendor implementation and network operator testing scope and interest from other vendors and operators is steadily growing.

This is where we noticed WT-508 at BBF its potential synergies.



### What we like about IETF YANG-Push

- Interoperable
- Unified with Netconf and Restconf
- Transport independent
- Push based

### What is unique to IETF YANG-Push

- Capabilities are discoverable which enables subscription automation
- Subscription state change notifications enables the discovery of schema changes
- YANG Library enables discovery of schema dependencies
- Standardized notification metadata

# IETF YANG-Push Implementations and Next Steps

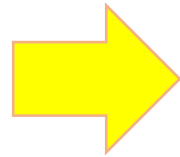
## Challenges and how to solve...

**What in IETF YANG-Push could have been defined differently and why?**

**What prevents IETF YANG-Push for being integrated/used efficiently?**

**What in IETF YANG-Push is missing and for which purpose?**

- On-change notification schema different than periodical
- Patch-id in On-Change complex to implement.
- Reduce YANG complexity (example: augmentations, deviations, xpath, lists)
- Each subscribed xpath needs normalization. High effort with many vendor specific YANG modules
- Missing end to end open-source implementations



- Extensible YANG-Push header combining notification and subscription. Separation of header and subscribed content is needed to allow partial parsing of message in binary encoding for data processing chain.
- On-change and periodical notification schema should have identical schema and contain the entire schema tree below subscription and represent current state.
- Common alignment on what should be supported in xpath and what not.
- **Agile incremental driven development.** Deployment guide describing implementers and operators what is/should be supported at which MVP stage.



# IETF YANG-Push Implementations and Next Steps

Incremental development...

## MVP 1 - Works

- [draft-ietf-netconf-udp-notif](#) transport (with segmentation option) and notifications encoded in RFC 7951 IETF-JSON.
- [RFC 8641](#) periodic subscription with anchor-time configurable in ietf-subscribed-notifications.
- YANG notifications as defined in [draft-netana-netconf-notif-envelope](#), YANG module name, version and yang-library-content-id as in [draft-ietf-netconf-yang-notifications-versioning](#) and notification capabilities ([RFC 9196](#)) discoverable as defined in their documents.
- YANG-Library as defined in [RFC 8525](#) and [draft-ietf-netconf-yang-library-augmentedby](#).

## MVP 2 – Scales and Secures

- Distributed notifications as defined in [draft-ietf-netconf-distributed-notif](#).
- YANG notifications encoded in CBOR (named identifiers) as defined in [RFC 9254](#).
- YANG notifications DTLS (1.2 MUST [RFC 6347](#), 1.3 SHOULD [RFC 9147](#)) encrypted as defined in [draft-ietf-netconf-udp-notif](#).

## MVP 3 - Optimizes

- [RFC 8641](#) on-change subscription configurable in ietf-subscribed-notifications.
- [RFC 8641](#) on-change and periodical subscription in [RFC 9196](#) defined capabilities discoverable.
- [draft-ietf-netconf-udp-notif](#) transport, encoding and encryption capabilities ([RFC 9196](#)) discoverable with [draft-netana-netconf-yp-transport-capabilities](#).

# An Architecture for YANG-Push to Message Broker Integration


## IETF 122 Hackathon

### Validate Configured Subscription YANG-Push Publisher Implementations

#### ► Champion(s)

Thomas Graf (thomas.graf @ [swisscom.com](mailto:thomas.graf@swisscom.com) )  
 Yannick Buchs (yannick.buchs @ [swisscom.com](mailto:yannick.buchs@swisscom.com) )  
 Daniel Voyer (danvoyerwork @ [gmail.com](mailto:dانvoyerwork@gmail.com) )  
 Holger Keller (holger.keller @ [telekom.de](mailto:holger.keller@telekom.de) )  
 Rob Wilton (rwilton @ [cisco.com](mailto:rwilton@cisco.com) )  
 Benoit Claise ([benoit.claise@huawei.com](mailto:benoit.claise@huawei.com))  
 Qiufang Ma (maqiufang1 @ [huawei.com](mailto:maqiufang1@huawei.com) )  
 Jérémie Leska (jeremie.leska @ [6wind.com](mailto:jeremie.leska@6wind.com) )  
 Samuel Gauthier (samuel.gauthier @ [6wind.com](mailto:samuel.gauthier@6wind.com) )

#### ► Draft Specifications

<https://datatracker.ietf.org/doc/html/rfc8639>   
<https://datatracker.ietf.org/doc/html/rfc8641>   
<https://datatracker.ietf.org/doc/html/rfc9196>   
<https://datatracker.ietf.org/doc/html/draft-netana-netconf-notif-envelope>   
<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-yang-notifications-versioning>   
<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-udp-notif>   
<https://datatracker.ietf.org/doc/html/draft-ietf-netconf-distributed-notif>   
<https://datatracker.ietf.org/doc/html/draft-netana-netconf-yp-transport-capabilities> 


#### ► Project Info

Validates and verify 3 YANG-Push vendor implementations in the area of:

#### ► Subscription automation

Discover YANG-Push systems and notifications capabilities and configure periodical and on-change subscriptions with netconf.

#### ► Notification integration

Validate subscription state change and push-update and push-change-update notifications for [draft-ietf-nmop-yang-message-broker-integration](#)  integration.

#### ► Configured subscription transport integration

Validate [draft-ietf-netconf-udp-notif](#)  and [draft-ietf-netconf-distributed-notif](#)  packet format on the wire.

#### ► Repository

<https://github.com/network-analytics/ietf-network-analytics-document-status/tree/main/122/Hackathon> 

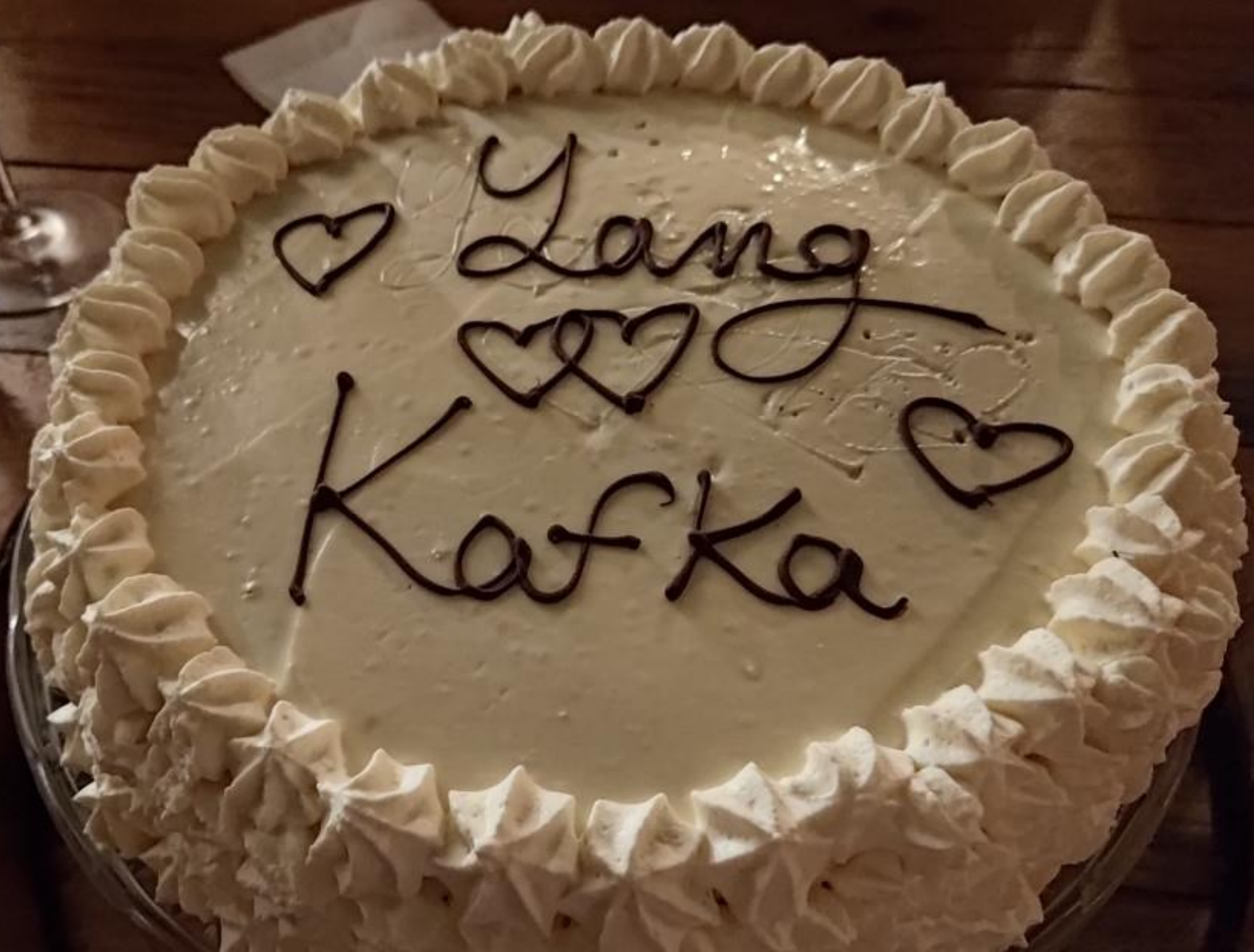
## YANG-Push Implementation Status

### IETF 121 – MVP 1

	6WIND VSR	Huawei VRP	Cisco IOS XR	Open- Source
RFC 8639 YANG-Push Subscription	✓	P	P	
RFC 8641 YANG-Push Notification	✓	P	✓	
draft-ietf-netconf-udp-notif	✓	✓	✓	✓
draft-ietf-netconf-yang-notifications-versioning	✓	✓	✓	
draft-tgraf-netconf-notif-sequencing	✓	✓	✓	
draft-tgraf-netconf-yang-push-observation-time	✓	✓	✓	
RFC 7895 YANG Library		✓		
RFC 8525 YANG Library (NMDA)	✓		✓	
draft-ietf-netconf-yang-library-augmentation		P		✓
RFC 9196 System and Notification Capabilities				
draft-netana-netconf-notif-envelope				

Green marked describes new capability at IETF 121. "P" to partially implemented.





♥ Yang ♥♥♥  
Kafka ♥