# YANG Message Keys for Message Broker Integration

draft-netana-nmop-yang-message-broker-message-key-01

A mechanism to define a unique message key for a
YANG to message broker integration and a topic addressing scheme
based on YANG-Push subscription type and a YANG index

thomas.graf@swisscom.com
ahmed.elhassany@swisscom.com
alex.huang-feng@insa-lyon.fr
benoit@everything-ops.net

26. October 2025

# We have a dream

## Digital Twin at your fingertips

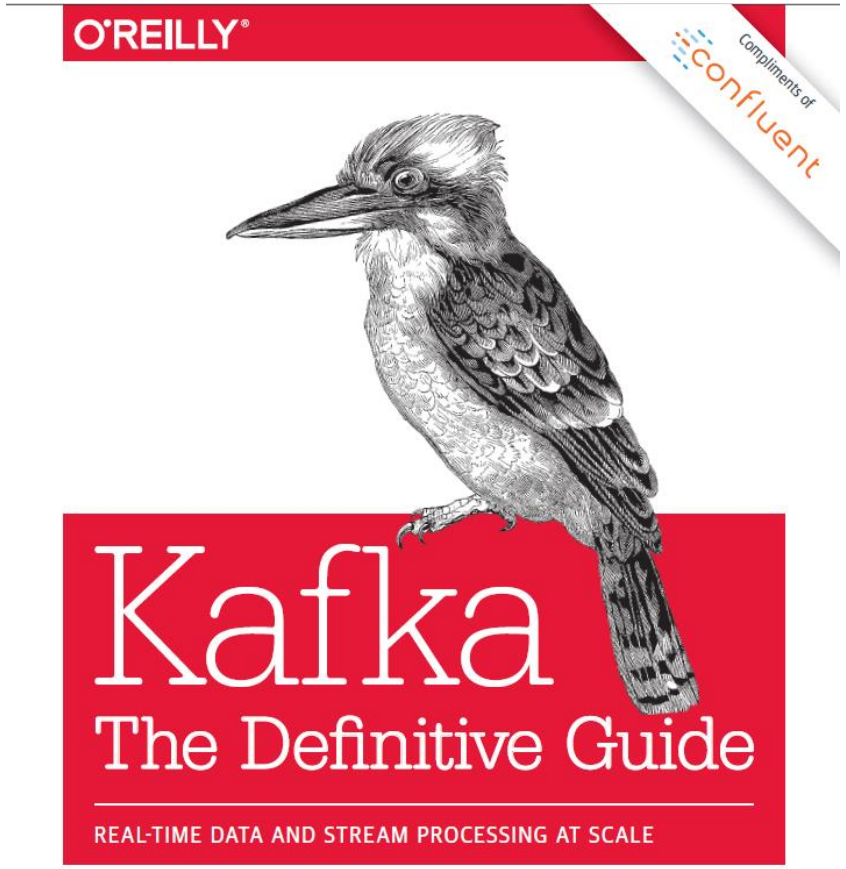
« Imagine that your entire life as network engineer you have logged into routers to perform show commands to get a glimpse into the current state of your networks. »

« Suddenly you see your colleague on the right querying the current network state in seconds directly from a real-time data stream. No access to routers needed. No databases needed. »

*"NASA Mission Control Digital Twin in the 60s, Digital Twin is not Rocket Science"*

# Terminology (1)
## Message Broker

- **Subject:** A named communication channel where a schema registry assigned schema id is associated.
- **Topic:** A communication channel for publishing and subscribing messages between producer and consumer with one or more subjects.
- **Topic Compaction:** The act of compressing messages in a topic to the latest state.
- **Partition:** Messages in a topic are spread over hash buckets where a hash bucket refers to a partition.
- **Message:** A piece of structured data sent between data processing components to facilitate communication in a distributed system
- **Message Key:** Metadata associated with a message to facilitate deterministic hash bucketing.
- **Segment:** A physical file containing multiple messages.




https://www.confluent.io/thank-you/resources/ebook/kafka-the-definitive-guide/

# Terminology (2)

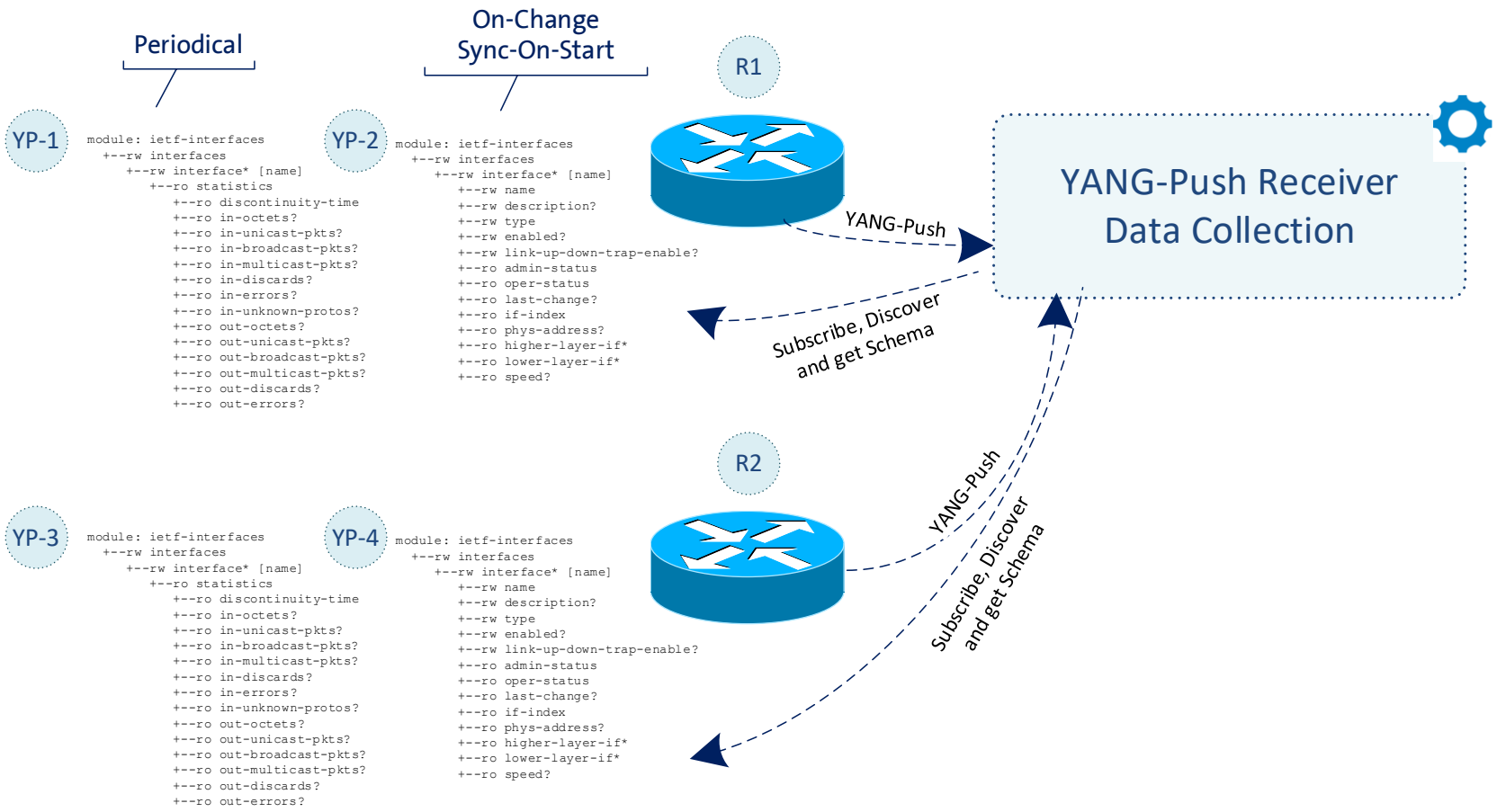
## YANG Data Taxonomy and Index

- **Dimensional Data:** Structured information in a data store [Ralph Kimball]. Example: YANG with data taxonomy.
- **Data Taxonomy:** Is a hierarchical system for classifying and organizing data into categories and subcategories
- **YANG Schema Tree:** The definition hierarchy specified within a module.
- **YANG Schema Node:** A node in the schema tree.
- **YANG Item Identifiers:** Section 3.3 in RFC 9254 defines unique text-based identifiers for each YANG element including YANG schema nodes.
- **YANG index:** Is a subset of YANG item identifiers containing only YANG schema node identifiers.

```
module: ietf-interfaces
     +--rw interfaces
        +--rw interface* [name]
           +--rw name                            string
           +--rw description?                    string
           +--rw type                            identityref
           +--rw enabled?                        boolean
           +--rw link-up-down-trap-enable?       enumeration {if-mib}?
           +--ro admin-status                    enumeration {if-mib}?
           +--ro oper-status                     enumeration
           +--ro last-change?                    yang:date-and-time
           +--ro if-index                        int32 {if-mib}?
           +--ro phys-address?                   yang:phys-address
           +--ro higher-layer-if*                interface-ref
           +--ro lower-layer-if*                 interface-ref
           +--ro speed?                          yang:gauge64
           +--ro statistics
              +--ro discontinuity-time    yang:date-and-time
              +--ro in-octets?            yang:counter64
              +--ro in-unicast-pkts?      yang:counter64
              +--ro in-broadcast-pkts?    yang:counter64
              +--ro in-multicast-pkts?    yang:counter64
              +--ro in-discards?          yang:counter32
              +--ro in-errors?            yang:counter32
              +--ro in-unknown-protos?    yang:counter32
              +--ro out-octets?           yang:counter64
              +--ro out-unicast-pkts?     yang:counter64
              +--ro out-broadcast-pkts?   yang:counter64
              +--ro out-multicast-pkts?   yang:counter64
              +--ro out-discards?         yang:counter32
              +--ro out-errors?           yang:counter32
```

# YANG-Push

## Discover and Subscribe to YANG metrics




From discovering YANG-Push subscription capabilities defined in RFC 9196, subscribing interesting metrics periodical (**statistics**), on-change (**state changes**) or on-change with sync-on-start (**states**) defined in RFC 8641.

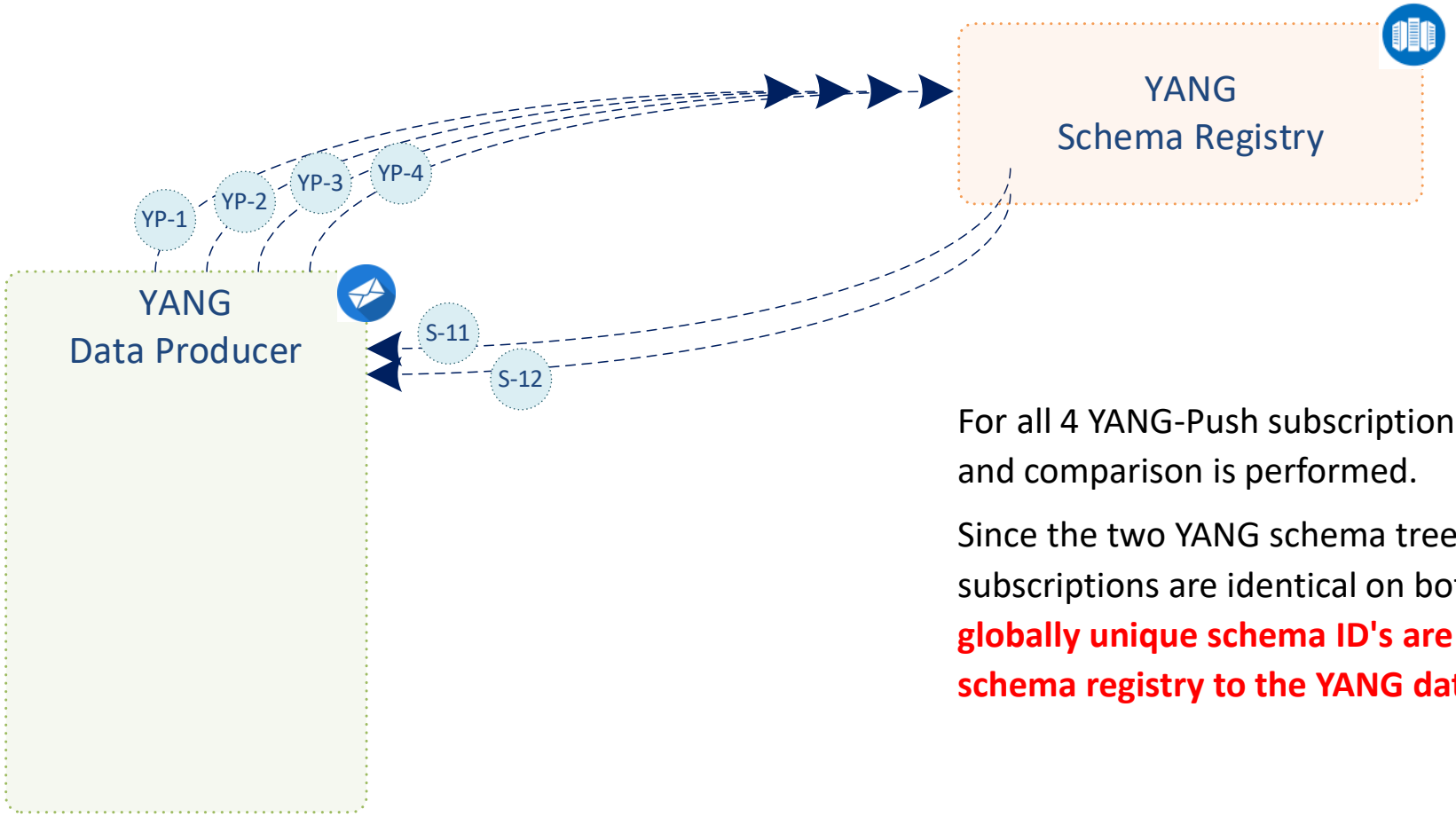
**Each subscription refers to network node, datastore (RFC 8342) and a schema tree.**

In this RFC 8343 example ietf-interface statistics are subscribed periodically and ietf-interface states on-change sync-on-start. **YANG-Push subscription ID's are per network node significant.**

Data Collection obtains for each subscription the YANG schema tree by leveraging <get-schema> (RFC 6022), YANG Library (RFC 8525) and draft-ietf-netconf-yang-library-augmentedby.

# YANG Schema Registry

## From 4 subscription ID's to 2 schema ID's

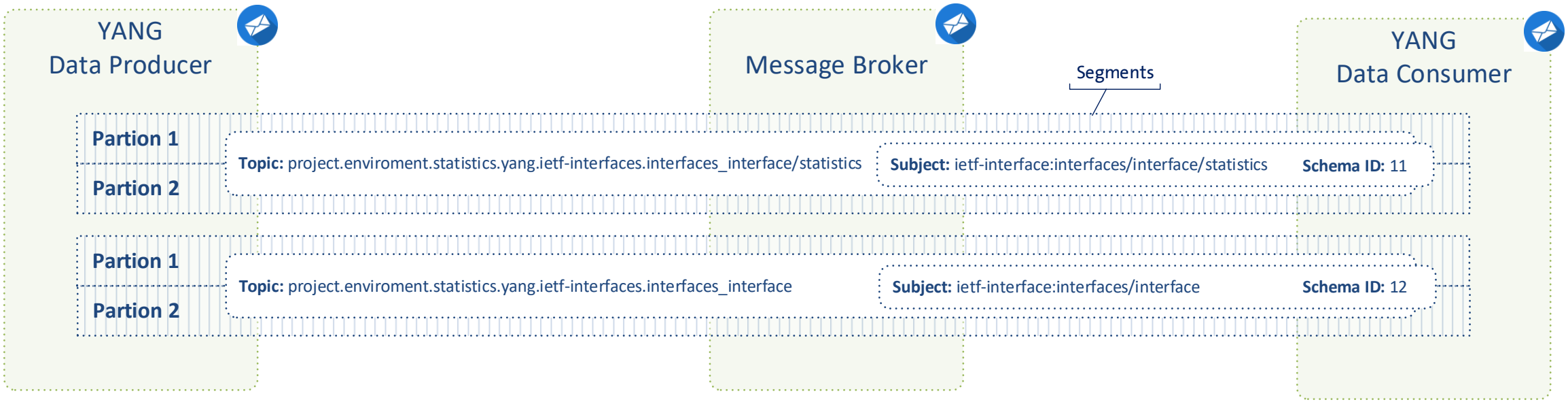



For all 4 YANG-Push subscriptions, YANG schema registration and comparison is performed.

Since the two YANG schema trees for both YANG-Push subscriptions are identical on both network nodes, **two YANG globally unique schema ID's are being issued from the YANG schema registry to the YANG data producer.**

# Message Broker

## Topics, Subjects, Partitions, Segments and Message Keys




**The YANG data producer creates for each YANG schema a new message broker topic, a message key and defines the number of partitions being used for the topic.**

It serializes the message with the previously generated message key and message content according to draft-ietf-nmop-message-broker-telemetry-message.

Each message is prefixed with the previously obtained schema ID representing a unique message subject. The messages are distributed according to the hashed message key across the partitions into continuous segments.

# YANG Message Keys and Indexes

## Calculated and Used at the YANG Data Producer

To calculate the YANG Index of the Message Key, the YANG item identifier needs to be extracted from the used YANG-Push subtree or xpath subscription filter. If the YANG item identifier is a YANG list as defined in Section 7.8 of RFC 7950 the YANG list key defined in Section 7.8.2 of RFC 7950 statement is suffixed with a "/" to the YANG item identifier.

For example, if the "ietf-interface:interfaces/interface[type='ianaift:ethernetCsmacd']" xpath filter is being used, the YANG item identifier is "ietf-interface:interfaces/interface". Interface is a YANG list with name as key. Therefore, the YANG Index of the Message Key is:

**ietf-interface:interfaces/interface/name**

When the Message is being produced to the Message Broker, the Network node hostname and YANG datastore name is used from the structured YANG data defined in "ietf-yang-push-telemetry-message" Section 3 of draft-ietf-nmop-message-broker-telemetry-message where the YANG Index is derived from subtree and xpath filters, respectively from their YANG schema tree.

# YANG-Push Message Broker Topic Naming

## Created at the YANG Data Producer

YANG can be subscribed periodically, on-change or on-change with sync-on-start. Periodical subscriptions are used for obtaining statistical metrics. On-Change subscriptions are used for obtaining State Changes and on-change with sync-on-start for obtaining States.

Message Brokers topics are addressed with a unique name. Usually, topics are named hierarchically similar to the DNS namespace where "." deliminates hierarchies.

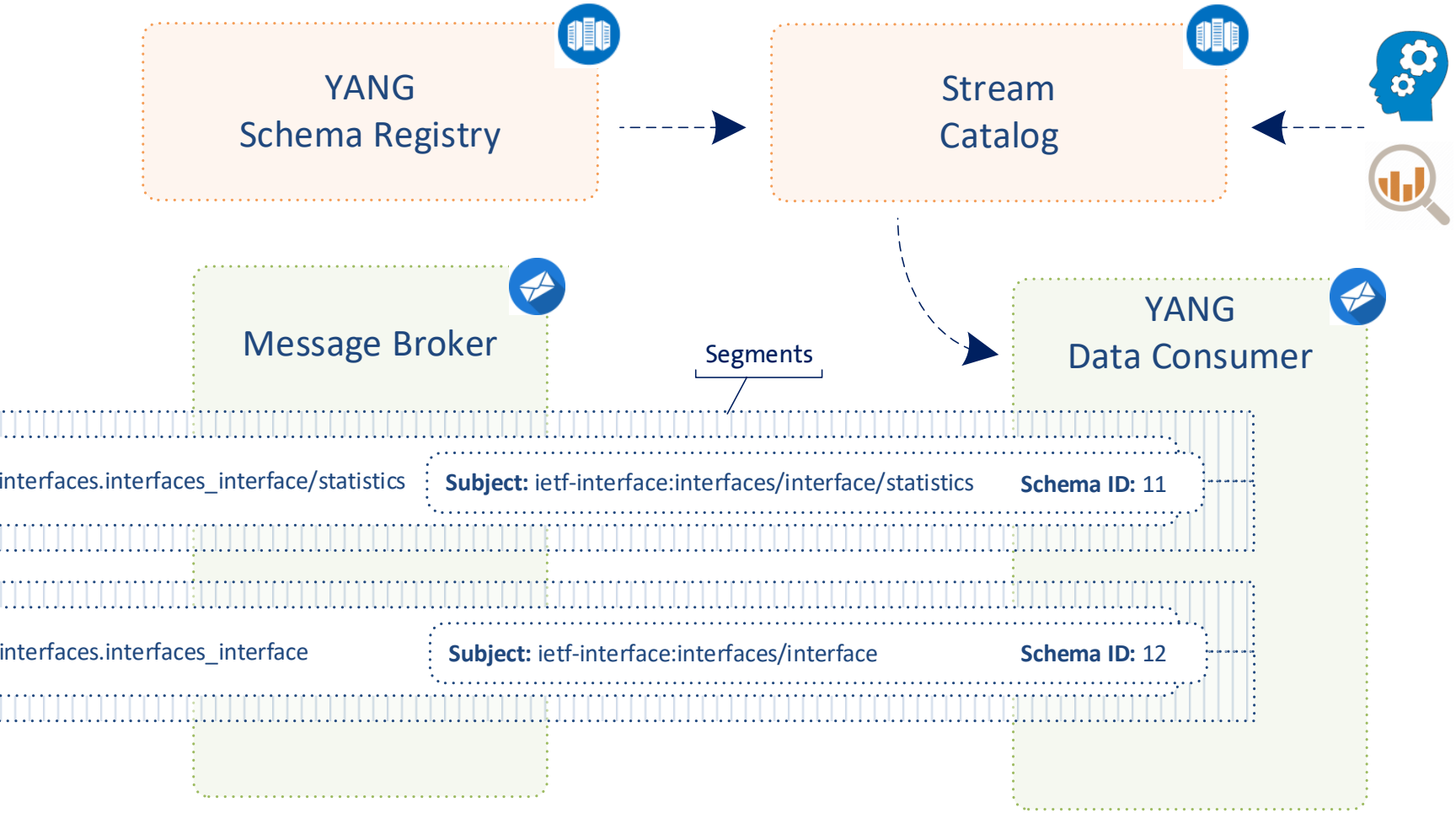
This document proposes to include "statistics", "states" and "state-changes" in the topic name as the first part to denote the types of data. Followed by "yang" to denote YANG data. Followed by the YANG module names subscribed and followed by the YANG Schema Node Identifier where "/" is substituted by "_".

For example, if the "ietf-interface:interfaces/interface" xpath filter is being used, the Message Broker topic name would be as following. In the example the project name and environment (prod, dev, test etc.) is prefixed:

**project.enviroment.statistics.yang.ietf-interfaces.interfaces_interface**

# YANG Data Consumption

## Discover and Subscribe to YANG metrics



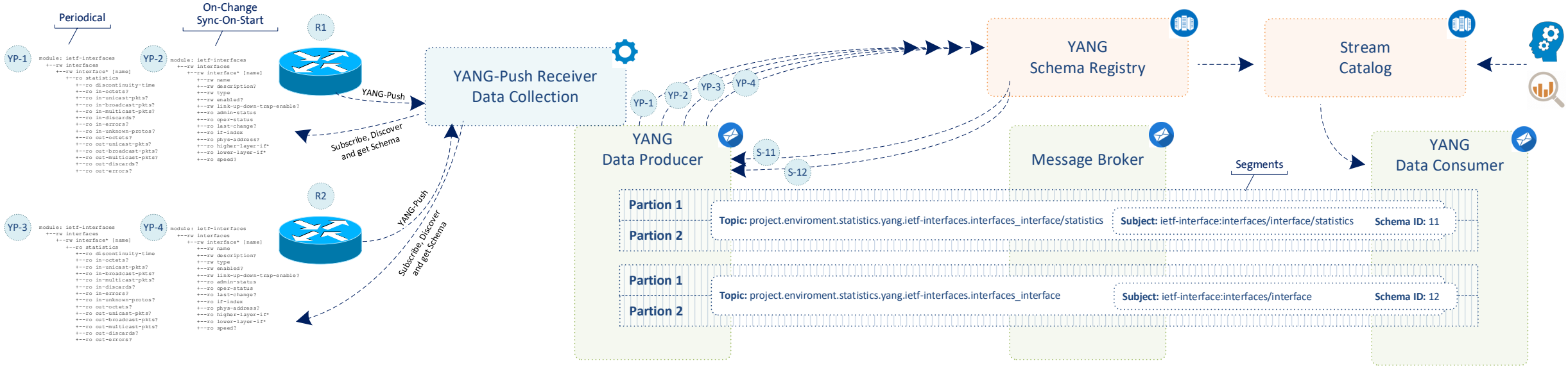

A user or AI application/agent subscribes discovers through the stream catalog interesting metrics and subscribes to message broker topic.

More than one topic can be consumed at once by using a wildcard such as: project.enviroment.states.yang.* to consume all YANG state metrics.

**The consumer hashes the message key and applies modulo with the number of partitions to determine the partition it needs to consume from to obtain messages with desired message key.**

# Fully YANG aware Data Pipeline

## From network nodes to message broker nodes




**At the YANG data producer,** thanks to the YANG awareness to the YANG data producer and the YANG schema registry, **for each unique YANG schema tree a message broker topic and subject is being created, and messages are being indexed and hashed according to the message key**.

**At the message broker,** thanks to the dedicated topic per YANG schema and its message key, **YANG state metrics can be compacted to the latest state; reducing the number of stored messages.**

**At the YANG data consumer,** thanks to the YANG awareness of the stream catalog and consumer, a specific topic, partition and subject can be consumed for the interested YANG metrics; **reducing the number of messages to be consumed.**

# Use Cases

## From network engineer to network controller to network management

**A network engineer** performs a SQL query on a set of message broker topics to obtain the current state and statistics of a given network dimension to verify a maintenance window or perform troubleshooting instead of logging into each network node to perform a show command.

**A network controller** performs a SQL query on a set of message broker topics to obtain the current state and statistics of a given network dimension to verify before and after the SRv6 traffic engineering configuration change instead of obtaining the metrics from the network. See: Section 3.8 of draft-ietf-nmop-simap-concept.

**A network management or network anomaly detection system** obtains the current YANG management plane state metrics from project.enviroment.states.yang.* for building a network inventory model and its state See: Section 3.2 from draft-ietf-nmop-simap-concept. From project.enviroment.statistis.yang.* it obtains statistics and maps them to the previously obtained network inventory model.

# Conclusion

## What we learned today

The differences between topics, partitions, subjects, segments, messages and message keys and how the apply to topic compaction.

That the following documents enabling many use cases, namely SIMAP and Network Anomaly Detection:

- draft-ietf-nmop-yang-message-broker-integration defines the YANG-Push to message broker integration architecture.
- draft-ietf-nmop-message-broker-telemetry-message defines the message schema being used between YANG data producer and consumer.
- draft-netana-nmop-yang-message-broker-message-key defines how YANG-Push metrics can be indexed and addressed efficiently and therefore reduce the number of consumed messages and reduce the number of stored metrics with topic compaction.

How YANG Message Keys and Indexes and YANG-Push Message Broker Topic Names are being generated.

That YANG-Push capabilities and YANG catalogs facilitate YANG metrics subscription in the network and data mesh.

# Network Digital Twin: Concepts and Reference Architecture

## draft-irtf-nmrg-network-digital-twin-arch

Section 6 of draft-irtf-nmrg-network-digital-twin-arch states:

```
Large-scale challenge:  A Digital Twin of large-scale networks will
      significantly increase the complexity of data acquisition and storage and the design and implementation of relevant
      models. The requirements of the software and hardware of the Network Digital Twin system will be even more constrained.
      Therefore, efficient and low cost tools in various fields should be required. Take data as an example, massive network
      data can help achieve more accurate models.  However, the cost of virtual-real communication and data storage becomes
      extremely expensive, especially in the multi- domain data-driven network management case, therefore efficient tools on
      data collection and data compression methods must be used.
```

- **The authors would like to hear from the NMRG working group wherever the proposed solution addresses this concern or not.**

Section 9.1 of draft-irtf-nmrg-network-digital-twin-arch states several Network Telemetry data collection methods but fails to describe its data mesh integration, how data should be organized, and maintained for a digital twin architecture.

- **Do you also agree that section 9.1 should be rewritten and the YANG-Push to Message Broker Integration Architecture should be described as one possible solution how Network Telemetry data collection should be collected and maintained for a digital twin architecture.**

thomas.graf@swisscom.com
ahmed.elhassany@swisscom.com
alex.huang-feng@insa-lyon.fr
benoit@everything-ops.net

26. October 2025


Yang
Kafka