

BMP YANG Model for Network Telemetry Messages

draft-netana-nmop-message-broker-bmp-telemetry-msg-02

A BGP Monitoring Protocol (BMP) message schema extension in YANG
to be used at data collection to transform Network Telemetry messages
into external systems such as Message Brokers

thomas.graf@swisscom.com
paolo@ntt.net
leonardo.rodoni@swisscom.com
maxence.younsi@insa-lyon.fr

14. March 2026

An Architecture for YANG-Push to Message Broker Integration

From YANG-Push, BMP and IPFIX to Network

A network operator aims for:

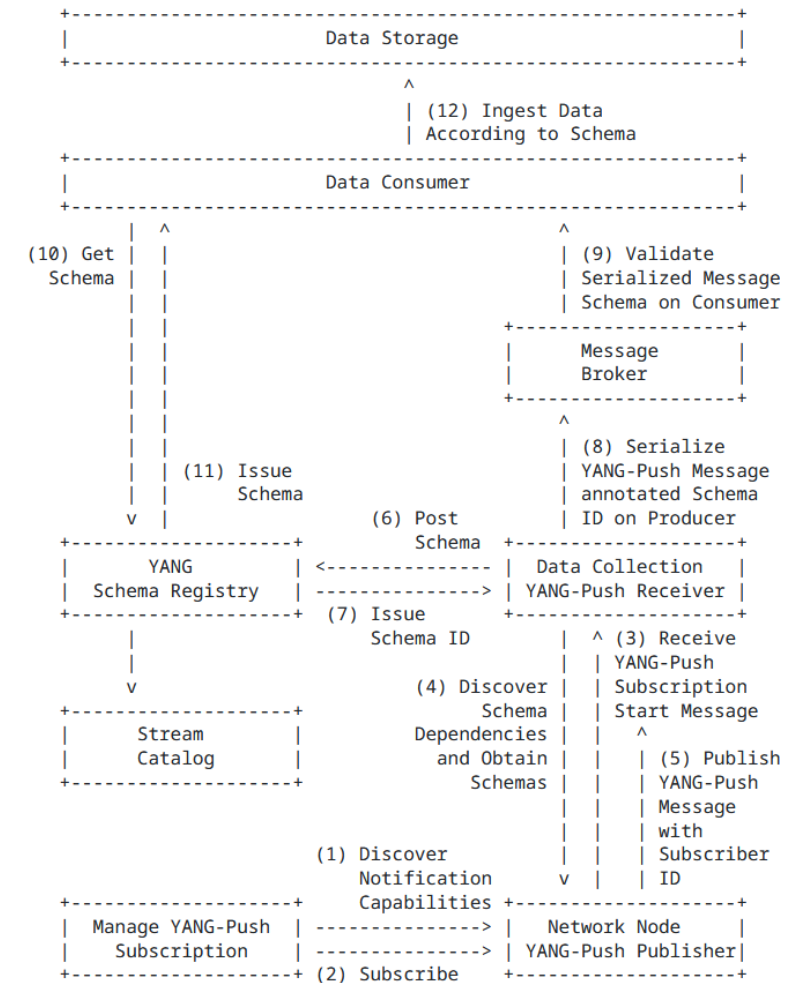
- An **automated data processing pipeline** which starts with YANG-Push, **BMP and IPFIX**, consolidates at Data Mesh **with YANG** and ends at Network Analytics.
- Operational metrics where **IETF defines the semantics**.
- Analytical metrics where **network operators gain actionable insights**.

These are the documents relate to the Message Broker integration:

- [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-ietf-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.
- **New:** [draft-netana-nmop-message-broker-bmp-telemetry-msg](#) defines the message schema extensions for BMP transformation and how message keys and topic names are derived.

4. Elements of the Architecture

The architecture consists of 6 elements. [Figure 1](#) gives an overview on the workflow.



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"*

BMP Monitors BGP

BGP RIB's with path decision and route-policy metadata

Integrates natively into the BGP process by mirroring BGP PDU's. Exports peering states and statistics and with route-monitoring the BGP RIB tables, BGP path decision and route-policy metadata.

BGP Monitoring Protocol (BMP)

<https://datatracker.ietf.org/doc/html/rfc7854>

Support for Adj-RIB-Out in BGP Monitoring Protocol

<https://tools.ietf.org/html/rfc8671>

Support for Local RIB in BGP Monitoring Protocol

<https://datatracker.ietf.org/doc/html/rfc9069>

TLV support for BMP Route Monitoring and Peer Down Messages

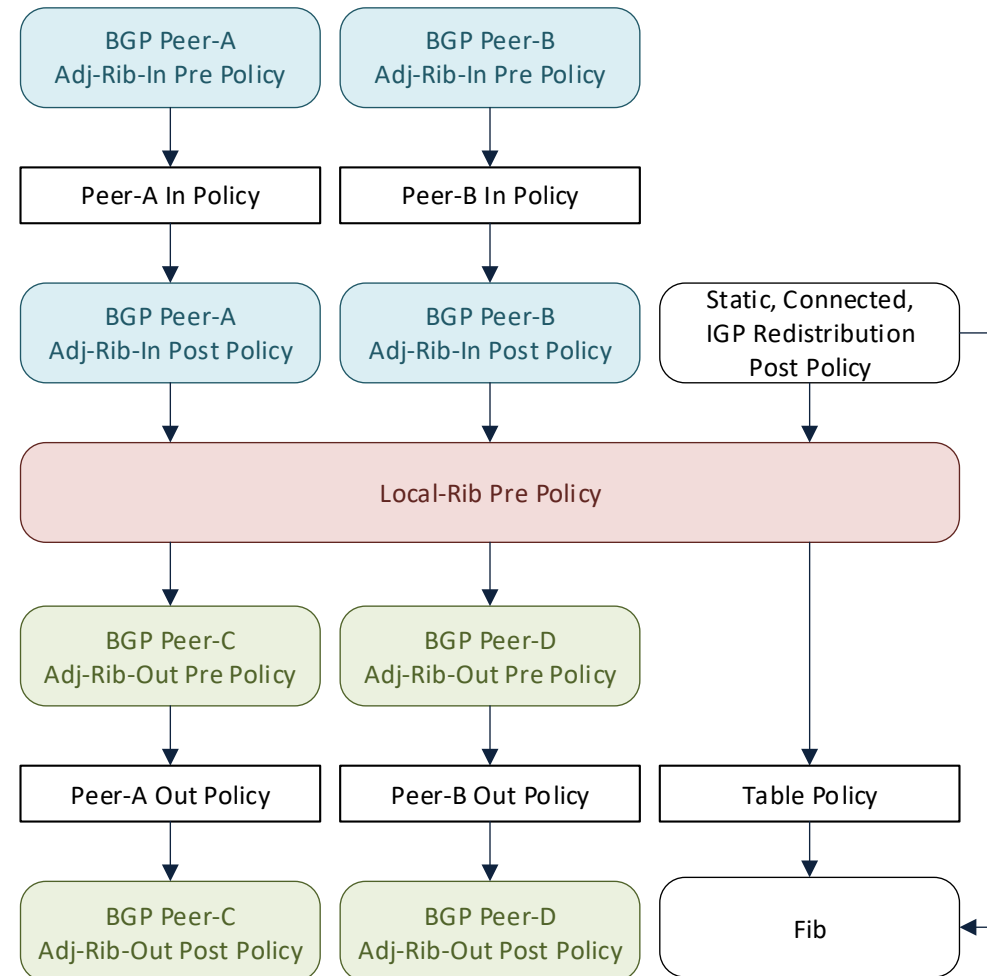
<https://tools.ietf.org/html/draft-ietf-grow-bmp-tlv>

BMP Extension for Path Marking TLV

<https://tools.ietf.org/html/draft-cppy-grow-bmp-path-marking-tlv>

Logging of routing events in BGP Monitoring Protocol

<https://datatracker.ietf.org/doc/html/draft-ietf-grow-bmp-rel>



BGP Monitoring Protocol

Address Family Agnostic

- BMP Per Peer Header**

Shows at **which RIB** (Adj-RIB In, Local or Adj-RIB Out, Pre or Post Policy) and from **which Peering** the BGP PDU at **which time** was obtained.

- Encapsulated BGP PDU**

Shows the encapsulated BGP PDU. In case of BMP route-monitoring, it describes whether it was a topology **update or withdrawal** and for **BGP community, NLRI and BGP Prefix SID** path attributes.

No.	Time	Source IP	Destination IP	Protocol	Length	Info
10	2023-11-06 22:12:33.943442	2001:db8:2::1	2a02:a90:4007::4:2	BGP	1294	UPDATE Message

```

> Frame 10: 1294 bytes on wire (10352 bits), 1294 bytes captured (10352 bits)
> Ethernet II, Src: Cisco_ff:dd:90 (40:06:d5:ff:dd:90), Dst: VMware_0e:db:14 (00:0c:29:0e:db:14)
> Internet Protocol Version 6, Src: 2001:db8:2::1, Dst: 2a02:a90:4007::4:2
> Transmission Control Protocol, Src Port: 39041, Dst Port: 1792, Seq: 746, Ack: 1, Len: 1220
▼ BGP Monitoring Protocol, Type Route Monitoring
  Version: 3
  Length: 227
  Type: Route Monitoring (0)
  ▼ Per Peer Header
    Type: Loc-RIB Instance Peer (3)
    > 0000 0000 = Flags: 0x00
    Peer Distinguisher: 0:0
    Unused: 000000000000000000000000
    Address: 0.0.0.0
    ASN: 65536
    BGP ID: 198.51.100.191
    Timestamp (sec): 1699272753
    Timestamp (msec): 942134
  ▼ Border Gateway Protocol - UPDATE Message
    Marker: ffffffffffffffffffffffffffffffff
    Length: 179
    Type: UPDATE Message (2)
    Withdrawn Routes Length: 0
    Total Path Attribute Length: 156
    ▼ Path attributes
      > Path Attribute - MP_REACH_NLRI
      > Path Attribute - ORIGIN: IGP
      > Path Attribute - AS_PATH: empty
      > Path Attribute - MULTI_EXIT_DISC: 0
      > Path Attribute - LOCAL_PREF: 16400
      > Path Attribute - COMMUNITIES: 64496:299 64496:1001 64497:1 64499:1
      > Path Attribute - EXTENDED_COMMUNITIES
      ▼ Path Attribute - BGP Prefix-SID
        > Flags: 0xc0, Optional, Transitive, Complete
        Type Code: BGP Prefix-SID (40)
        Length: 37
        ▼ SRV6 L3 Service
          Type: SRV6 L3 Service (5)
          Length: 34
          Reserved: 00
          ▼ SRV6 Service Sub-TLVs
            ▼ SRV6 Service Sub-TLV - SRV6 SID Information
              Type: SRV6 SID Information (1)
              Length: 30
              Reserved: 00
              SRV6 SID Value: 2001:db8:1::
              SRV6 SID Flags: 0x00
              SRV6 Endpoint Behavior: End.DT4 with NEXT-CSID (0x003f)
              Reserved: 00
            ▼ SRV6 Service Data Sub-Sub-TLVs
              ▼ SRV6 Service Data Sub-Sub-TLV - SRV6 SID Structure
                Type: SRV6 SID Structure (1)
                Length: 6
                Locator Block Length: 32
                Locator Node Length: 16
                Function Length: 16
                Argument Length: 0
                Transposition Length: 16
                Transposition Offset: 48
  
```

Message Broker Terminology

- **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.

O'REILLY®

Compliments of
confluent



Kafka The Definitive Guide

REAL-TIME DATA AND STREAM PROCESSING AT SCALE

Neha Narkhede,
Gwen Shapira & Todd Palino

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

YANG Data Terminology

- **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. [See RFC 8340](#).
- **YANG Data Node:** An instantiated node in the schema tree such as container, leaf, leaf-list, list, choice and case elements.

```
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
```

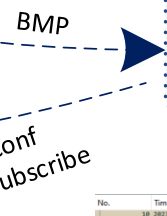
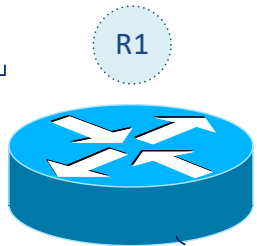
BGP Monitoring Protocol

Subscribe to BGP metrics

Subscribe to BMP
route-monitoring
and statistics

```

module: ietf-bmp
+--rw bmp
+--rw monitoring-stations
+--rw monitoring-station* [id]
  +--rw id string
  +--rw description? string
  +--rw connection
  +--rw (passive-or-active)
  +-- (active)
  +--rw active
    +--rw network-instance?
      leafref
    +--rw station-address
      inet:ip-address
    +--rw station-port
      inet:port-number
    +--rw (local-endpoint)
      |
      | +--rw monitored-router-address?
      | |
      | | +--rw monitored-router-address?
      | | |
      | | | +--rw monitored-router-interface?
      | | | |
      | | | | if:interface-ref
      | | | +--rw monitored-router-port?
      | | | |
      | | | | inet:port-number
  
```



```

module: ietf-bmp
+--rw bmp
+--rw monitoring-stations
+--rw monitoring-station* [id]
  +--rw id string
  +--rw description? string
  +--rw bmp-data
  +--rw initiation-message? string
  +--rw statistics-report!
    | +--rw statistics-interval uint32
  +--rw route-monitoring
  +--rw network-instance-configuration
  +--rw network-instances
  +--rw network-instance* [id]
    +--rw id leafref
    +--rw enabled? boolean
    +--rw local-rib
    +--rw address-families
    +--rw address-family* [id]
      +--rw id identityref
    +--rw filters
    +--rw policy-filter
      {bmp-filter-based-on-route-policy?}
    +--rw export-policy*
      | leafref
    +--rw default-export-policy?
      rt-pol:default-policy-type
  
```

No.	Time	Source IP	Destination IP	Protocol	Length	Info
1	19 2023-11-06 22:12:33.343442	2001:000:12:1:1	2002:090:0007:14:2	BGP	1294	UPDATE MESSAGE

```

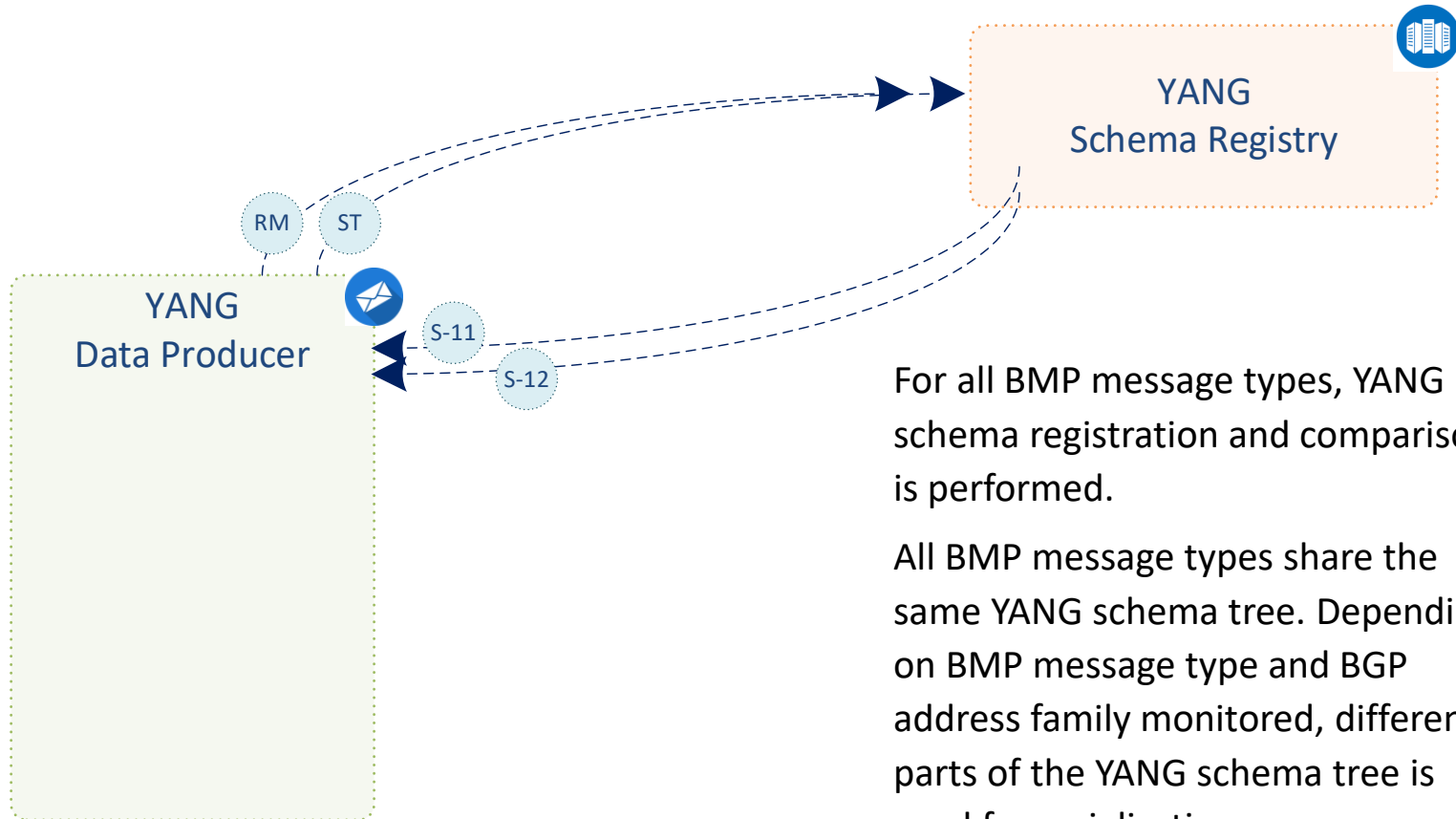
Frame 18: 1294 Bytes on wire (38932 bits), 1294 bytes captured (38932 bits)
> Ethernet II, Src: Cisco_Fi0E/90 (08:00:0E:FF:00:90), Dst: VMware_M010R14 (08:00:C:29:00:00:14)
> Internet Protocol Version 6, Src: 2001:000:12:1:1, Dst: 2002:090:0007:14:2
> Transmission Control Protocol, Src Port: 39041, Dst Port: 1791, Seq: 746, Ack: 1, Len: 1220
> BGP Monitoring Protocol, Type Route Monitoring
  version: 3
  Length: 227
  Type: Route Monitoring (0)
  Per Peer header
  Type: Loc-RIB Instance Peer (3)
  > none none = flags: none
  Peer Distinguisher: 0:0
  Local: 00000000000000000000000000000000
  Address: 0.0.0.0
  ASN: 65536
  BGP ID: 190.15.100.101
  Timestamp (sec): 1699272793
  Timestamp (msec): 94214
  > Border Gateway Protocol - UPDATE Message
  Marker: 00000000000000000000000000000000
  Length: 170
  Type: UPDATE Message (2)
  Withdrawn Routes Length: 0
  Total Path Attribute Length: 156
  Path attributes
  > Path attribute - MP_RESECURBLAS
  > Path attribute - ORIGIN: 000
  > Path attribute - AS_PATH: 0000
  > Path attribute - MULTI_EXIT_DISC: 0
  > Path attribute - LOCAL_PREF: 16408
  > Path attribute - COMMUNITIES: 64496:129 64496:1292 64497:1 64499:1
  > Path attribute - EXTENDED_COMMUNITIES
  > Path attribute - BGP Prefix-SID
  > Flags: Next, Optional, Transitive, Complete
  Type Code: BGP Prefix-SID (48)
  Length: 12
  > Srv6 L3 Service
  Type: Srv6 L3 Service (5)
  Length: 34
  Reserved: 00
  > Srv6 Service Sub-TLVs
  > Srv6 Service Sub-TLV - Srv6 SID Information
  Type: Srv6 SID Information (1)
  Length: 18
  Reserved: 00
  Srv6 SID Value: 2001:000:12:1:1
  Srv6 SID Flags: 0000
  Srv6 Endpoint Behavior: end.01a with NEXT-CSID (00000F)
  Reserved: 00
  > Srv6 Service Data Sub-TLVs
  > Srv6 Service Data Sub-TLV - Srv6 SID Structure
  Type: Srv6 SID Structure (1)
  Length: 6
  Locator Block Length: 32
  Locator Mode Length: 36
  Function Length: 16
  Argument Length: 8
  Transposition Length: 16
  Transposition Offset: 48
  
```

Through Netconf/Restconf and [draft-ietf-grow-bmp-yang](#) defined YANG modules BMP metrics are subscribed. With BMPv3 defined in [RFC 7854](#) resp. [RFC 8671](#) and [RFC 9069](#) information, peer-up, peer-down, route-monitoring, route-mirroring and statistic messages are exported to BMP monitoring station.

At the BMP monitoring station, the received BMP messages are **transformed to YANG** according to [draft-netana-nmop-message-broker-bmp-telemetry-msg](#) defined YANG modules.

YANG Schema Registry

Register BMP Telemetry Message Schemas



For all BMP message types, YANG schema registration and comparison is performed.

All BMP message types share the same YANG schema tree. Depending on BMP message type and BGP address family monitored, different parts of the YANG schema tree is used for serialization.

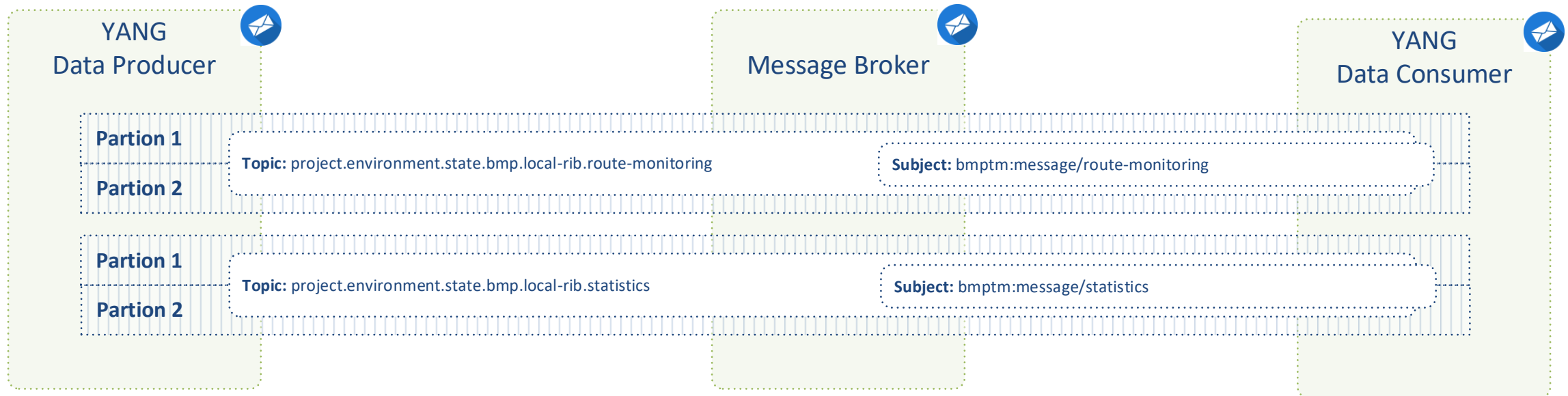
```

module: ietf-bmp-telemetry-message

structure message:
  +-- version?          uint8
  +-- (message-type)?
  +--:(route-monitoring)
  +-- route-monitoring
  +-- peer-type?       peer-type
  +-- peer-flags?      uint8
  +-- peer-distinguisher? rt-types:route-distinguisher
  +-- peer-address?    inet:ip-address
  +-- peer-as?         uint32
  +-- timestamp?      yang:date-and-time
  +-- afi-safi-type    identityref
  +-- rib-entry
  +-- (afi-safi)
  +--:(ipv4-unicast)
  +-- ipv4-unicast
  +-- (rib-type)
  +--:(loc-rib)
  +-- loc-rib
  +-- route
  +-- prefix            inet:ipv4-prefix
  +-- origin?          union
  +-- path-id?         uint32
  +-- attributes
  | +-- origin?        bt:bgp-origin-attr-type
  | +-- as-path
  | | +-- segment*
  | | | +-- type?      identityref
  | | | +-- member*    inet:as-number
  | | +-- next-hop?   inet:ip-address
  | | +-- link-local-next-hop? inet:ipv6-address
  | | +-- med?         uint32
  | | +-- local-pref? uint32
  | | +-- as4-path
  | | +-- segment*
  | | | +-- type?      identityref
  | | | +-- member*    inet:as-number
  | | +-- aggregator
  | | | +-- as?         inet:as-number
  | | | +-- identifier? yang:dotted-quad
  | | +-- aggregator4
  | | | +-- as4?       inet:as-number
  | | | +-- identifier? yang:dotted-quad
  | | +-- atomic-aggregate? boolean
  | | +-- originator-id? yang:dotted-quad
  | | +-- cluster-list* yang:dotted-quad
  | | +-- aigp-metric?  uint64
  | +-- community*    union
  | | +-- ext-community* bct:bgp-ext-community-type
  | | +-- ext-community-raw* string
  | | +-- ipv6-ext-community* bct:bgp-ipv6-ext-community-type
  | | +-- ipv6-ext-community-raw* string
  | | +-- large-community* bct:bgp-large-community-type
  | | +-- last-modified? yang:timeticks
  | | +-- eligible-route? boolean
  | | +-- ineligible-reason? identityref
  | +-- unknown-attributes
  | | +-- unknown-attribute* [attr-type]
  | | | +-- attr-type    uint8
  | | | +-- optional?   boolean
  | | | +-- transitive? boolean
  | | | +-- partial?    boolean
  | | | +-- extended?   boolean
  | | | +-- attr-len?   uint16
  | | | +-- attr-value? binary
  +-- reject-reason?  union
  
```

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](#) and [draft-netana-nmop-message-broker-bmp-telemetry-msg](#) augment.

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.

BMP YANG Message Broker Topic Naming

Created at the YANG Message Broker Producer

BMP data can be subscribed for BMP session and BGP peering session state-changes, BGP peering statistics or BGP RIB states. as defined in [Section 3.1 of RFC 7854](#).

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

The document defines "statistics", "states" and "state-changes" in the topic name as the first part to denote the types of data. Followed by "bmp" to denote BMP data. Followed by the BGP RIB type and the BMP message type name.

For example, if BMP route-monitoring is being subscribed to BGP Local-RIB, the Message Broker topic name would be as following. In the example the project name and environment (prod, dev, test etc.) is prefixed.

project.environment.state.bmp.local-rib.route-monitoring

YANG Message Keys and Indexes

Calculated and Used at the YANG Message Broker Producer

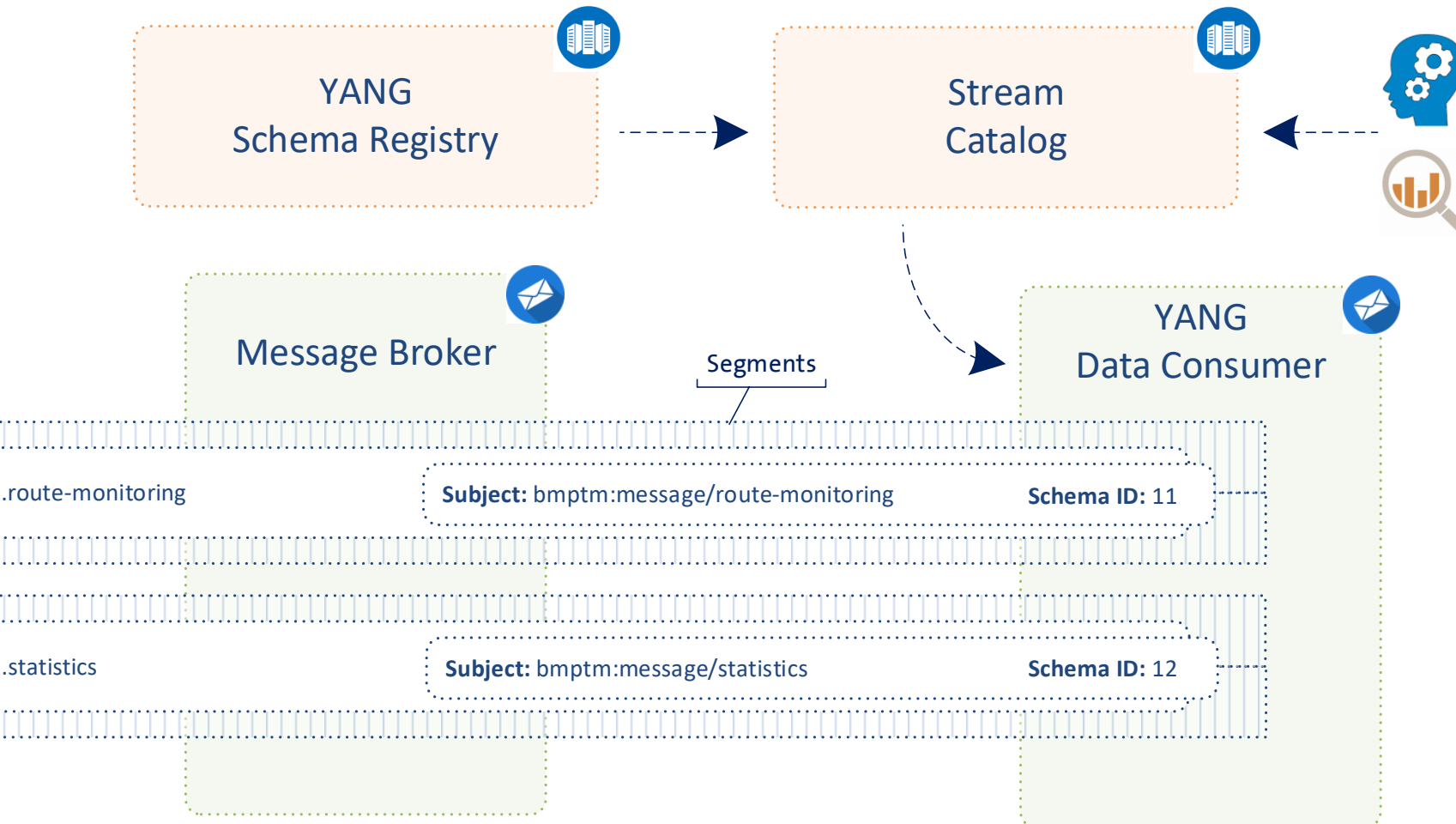
A Message Broker uses a Message Key to index the Message and a value to carry the Message content. **The Message Key is being used as input for the Message Broker Producer hash function to distribute across Partitions and at the Message Broker for topic compaction.**

For BMP, **the network node hostname, what BMP Message Type the data represents and** apart from the initiation-message and termination-message **the BMP index** is used to generate the Message Key. The BMP index is calculated depending on BMP Message Type as following:

- peer-up-notification: "peer-distinguisher" and "peer-address".
- peer-down-notification: "peer-distinguisher" and "peer-address".
- statistics: "peer-distinguisher" and "peer-address".
- route-monitoring: "peer-distinguisher", "peer-address" afi-safi name, address family, name of the BGP RIB, "prefix" "origin" "path-id" route attributes and the index of the used attr-set.
- route-mirroring: "peer-distinguisher", "peer-address" afi-safi name, address family, name of the BGP RIB, "prefix" "origin" "path-id" route attributes, index of the used attr-set and information enumeration.

BMP YANG Data Consumption

Discover and Subscribe to BMP 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.environment.state.bmp.local-rib.* to consume all BMP Local-RIB 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.

Use Cases

From network engineer to network controller to network management

A network engineer performs a SQL query on "project.environment.state.bmp.adj-rib-in-post.*" and "project.environment.statistics.bmp.adj-rib-in-post.*" message broker topics to obtain the current state and state-change for a set of BGP peerings 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 "project.environment.state.bmp.local-rib.*" and "project.environment.statistics.bmp.local-rib.*" to obtain the current state and statistics of a given BGP path 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 BMP Local-RIB state from "project.environment.state.bmp.local-rib.*" for building a network topology. See: [Section 3.2 from draft-ietf-nmop-simap-concept](#). From "project.environment.statistics.bmp.local-rib.*" it obtains statistics and maps them to the previously obtained network topology.

Questions to...

NMOP and GROW

- Do you resonate of having BMP and IPFIX data being transformed to YANG and integrated uniformly to Message Broker to ease data consumption for network operation and next generation AI applications?
- YANG-Push Message Broker integration has its home at NMOP, BMP at GROW and IPFIX at OPSAWG. Do you agree that having all Message Broker documents in NMOP regardless which Network Telemetry data is being collected and transformed would ease conformity?
- The document introduces "ietf-bmp-bgp-rib-entry". Compared to "ietf-bgp-rib" that structure supports that each BGP path can be streamed individually or as JSON array in a message broker message. Do you agree that this should be defined in this document and groupings reused from "ietf-bgp-rib-attributes" and "ietf-bgp-rib-tables"?

Questions to...

draft-ietf-idr-bgp-model authors and IDR

- [Section 3.2 of draft-netana-nmop-message-broker-bmp-telemetry-msg](#) describes the relationship to [draft-ietf-idr-bgp-model](#) and proposes changes to "ietf-bgp-rib-attributes" and "ietf-bgp-rib-tables" YANG modules. The draft-netana-nmop-message-broker-bmp-telemetry-msg authors ask feedback from the draft-ietf-idr-bgp-model authors and IDR working group on document status, intended timeline and wherever proposed changes can be incorporated or not.
- [draft-ietf-idr-bgp-model](#) states that the model supports ipv4-unicast and ipv6-unicast address-families and defers the remaining AFI/SAFI to other or future drafts. [draft-netana-nmop-message-broker-bmp-telemetry-msg](#) authors intend to cover l3vpn-ipv4-unicast, l3vpn-ipv6-unicast, l2vpn-evpn and ipv4-labeled-unicast for covering operational BGP RIB. The current understanding is that this has not been addressed at the IETF yet correct?

Next Steps

Extend model, working group adoption and code contribution

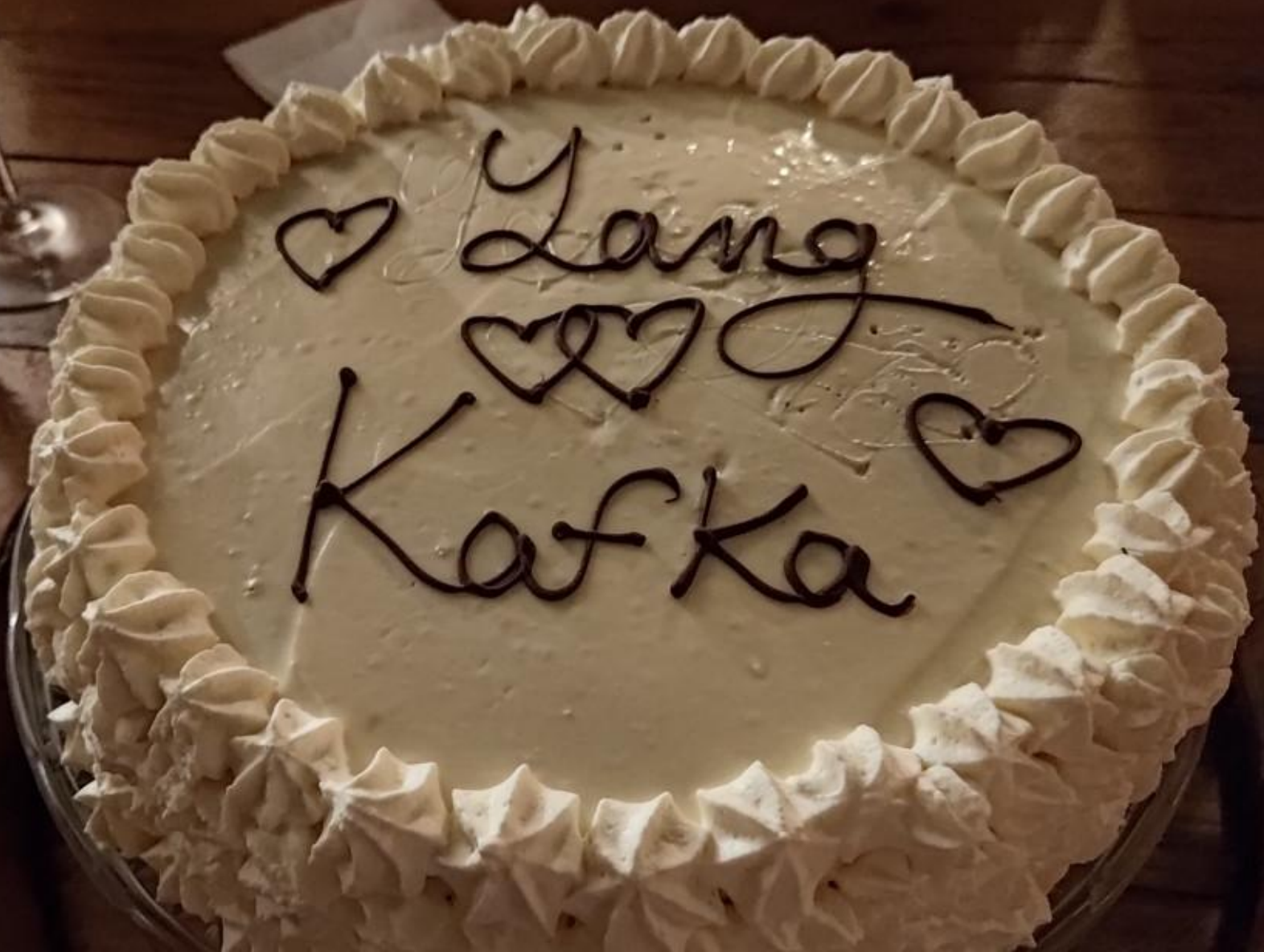
- Extend document to cover:
 - BGP Open Message from [Section 4.2 of RFC 4271](#) and its BGP Capabilities from [RFC 5492](#) in BMP peer-up message type by leveraging ietf-bgp-capabilities@2024-10-21.yang from [draft-ietf-idr-bgp-model](#).
 - BMP Information TLV from [Section 4.4 of RFC 7854](#) in BMP peer-up message type.
 - BMP Group TLV, Stateless TLV and VRF TLV from [draft-ietf-grow-bmp-tlv](#) and BGP Best Path metadata from [draft-ietf-grow-bmp-path-marking-tlv](#)
 - VRF Name and ID for peer-up and route-monitoring message types from [Section 5.2.1 of RFC 9069](#)
 - Additional BGP address families
- Request NMOP, GROW and IDR working group feedback and adoption at NMOP or GROW.
- Contribute to [NetGauze](#) BMP data collection and YANG Message Broker Producer.

Conclusion

What we learned at IETF 124 and now at IETF 125

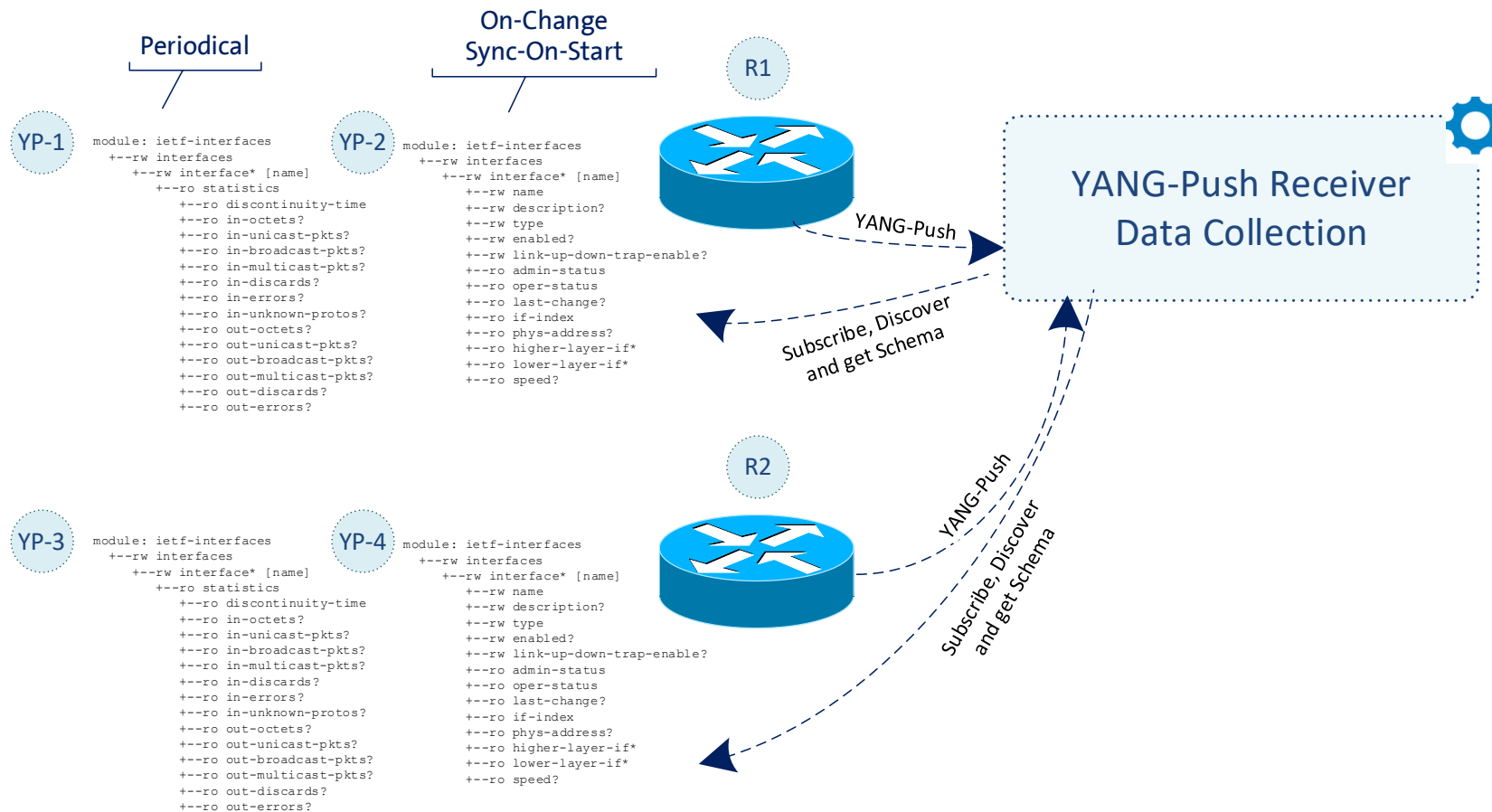
- At IETF 124 we introduced NMOP to the differences between topics, partitions, subjects, segments, messages and message keys and how they apply to topic naming and compaction and how YANG-Push capabilities and YANG catalogs facilitate YANG metrics subscription in the network and data mesh.
- At this IETF 125 we introduced NMOP and GROW to the YANG models for subscribing and transforming BMP data and their relationship. How topics are named and indexed based on BMP message types. There are action items on modelling BGP and BMP data in YANG.

...and BMP and soon
also IPFIX as well



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](#).

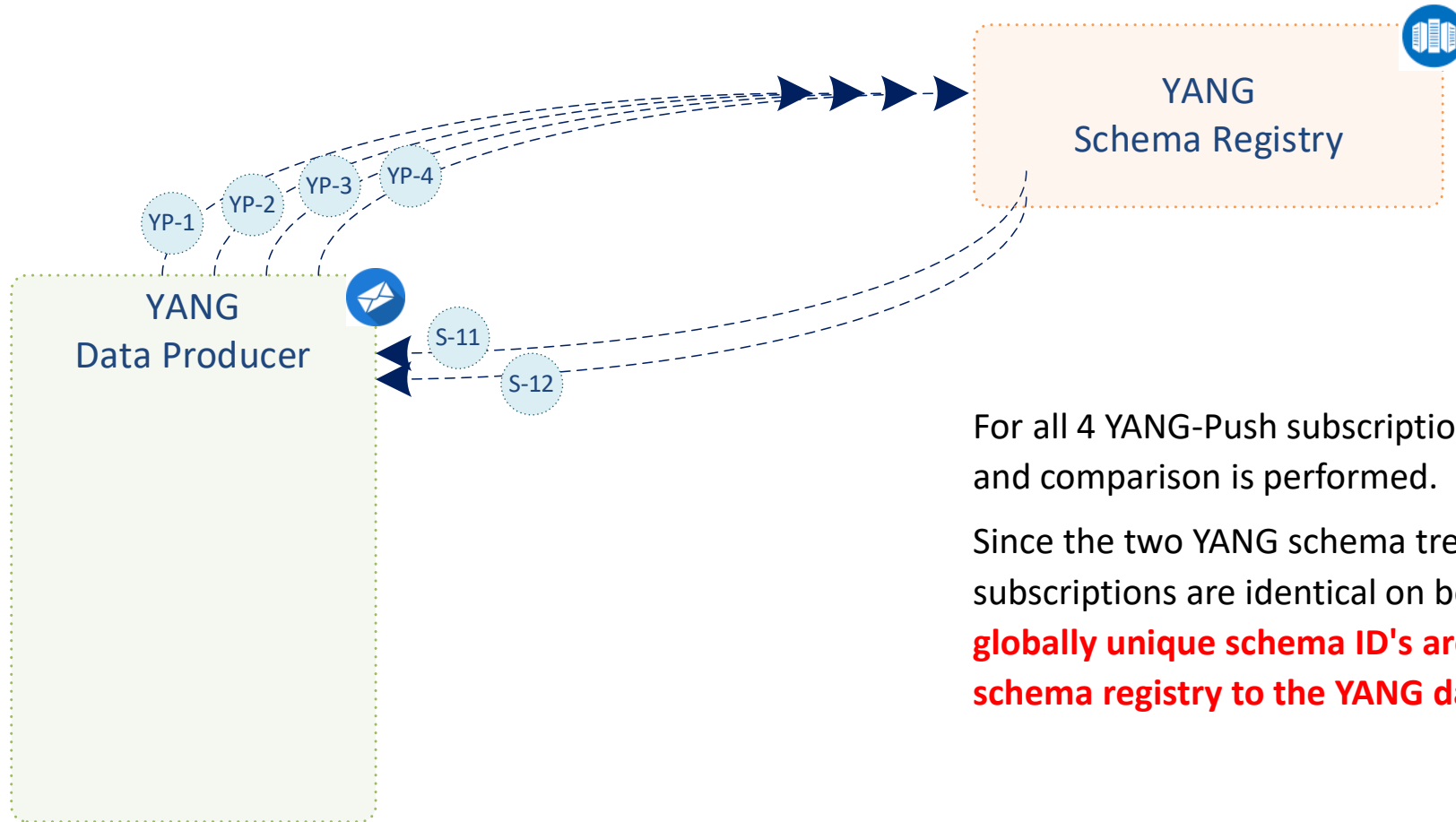
Each subscription refers to network node, datastore ([RFC 8342](#)) and a schema tree.

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

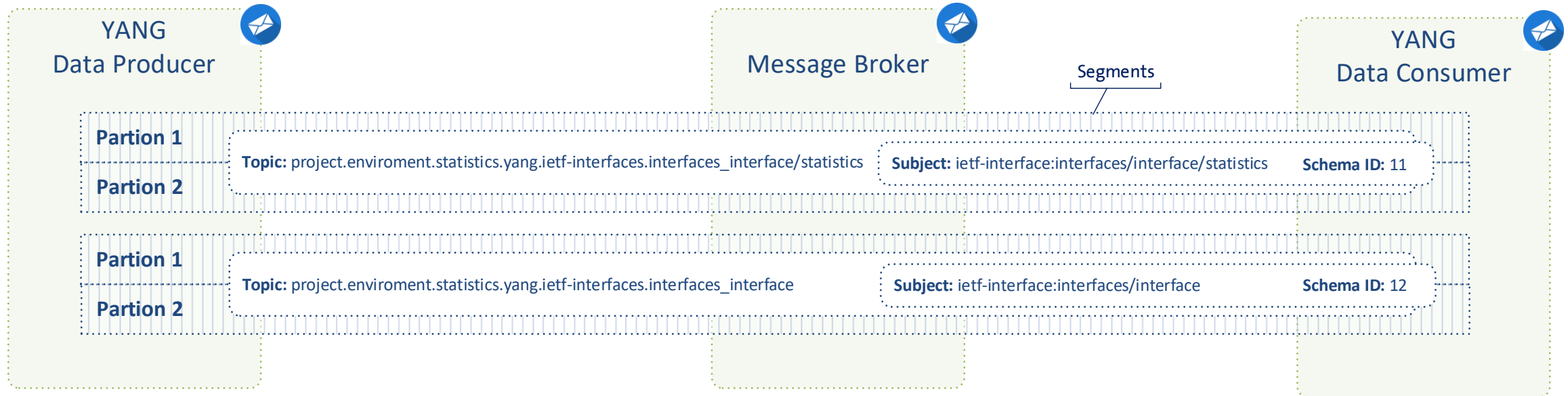


For all 4 YANG-Push subscriptions, YANG schema registration and comparison is performed.

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 "." delaminates 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