IETF 106 – Singapore
November 2019

# SR-Policy YANG

draft-raza-spring-sr-policy-yang-02

K. Raza (Cisco)

R. Sawaya (Cisco)

Z. Shunwan ((Huawei)

D. Voyer (Bell Canada)

M. Durrani (Equinix)

S. Matsushima (SoftBank)

V. Beeram (Juniper)

# Overview

- YANG data model for SR Traffic-Engineering Policies
  - Configure, instantiate, and manage
- The model is:
  - Aligned with SR-policy architecture doc (draft-ietf-spring-segment-routing-policy)
  - Generic and apply equally to the SR-MPLS and SRv6 instantiations of SR policies

# Modules

- ietf-sr-policy-types:
  - defines common and basic types related to an SR policy and related constructs
- ietf-sr-policy:
  - defines the model for SR policy configuration and management

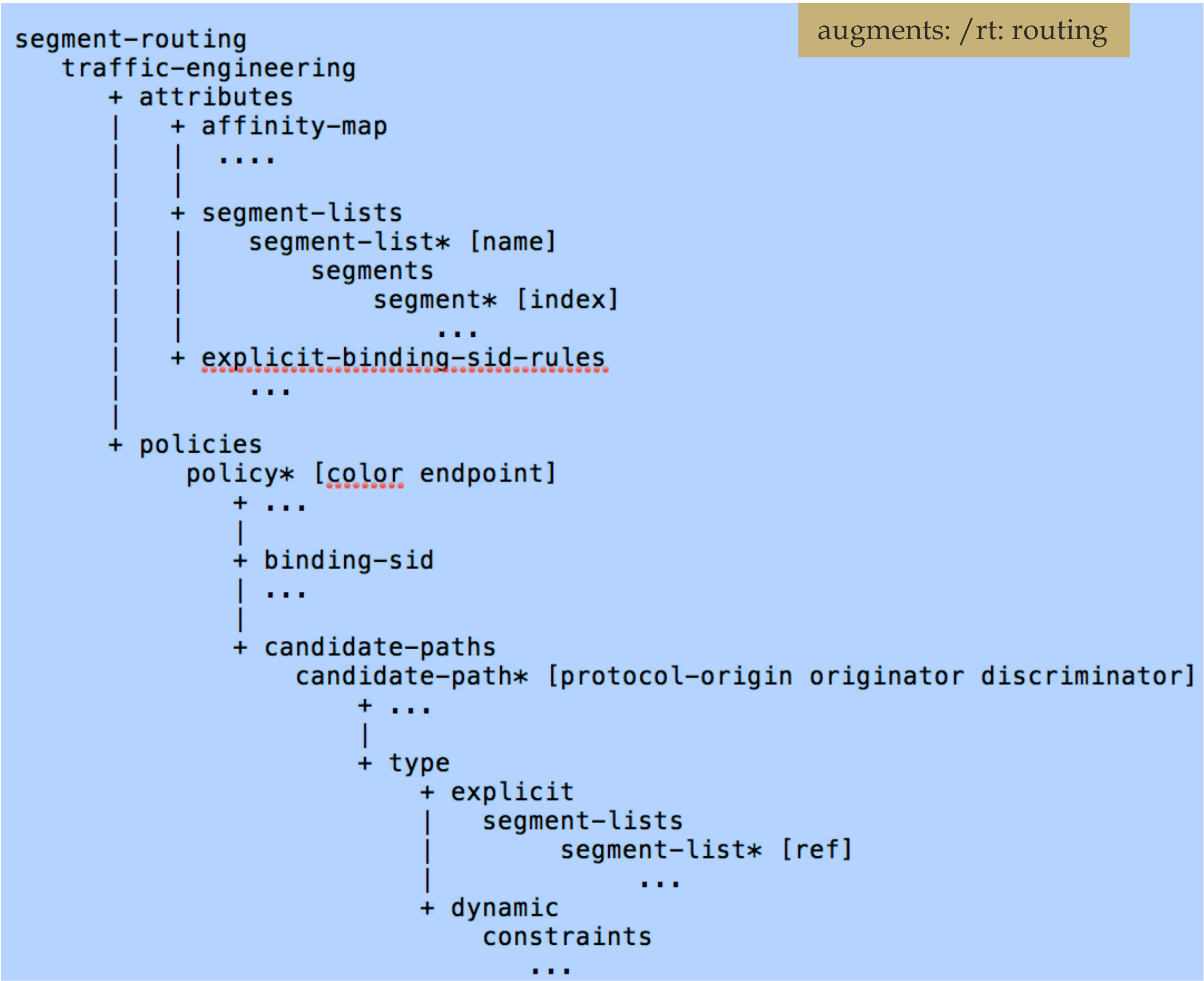
# ietf-sr-policy-types

- dataplane-type: MPLS or SRv6
- sid-value-type: Label or IPv6 SID
- segment-type
- path-disjointness: link, node, ..
- binding-sid-alloc-mode: Dynamic vs Explicit
- explicit-binding-sid-rule-type:  BSID Explicit alloc mode rules/enforcements
- Protocol-origin-type: CLI, PCEP, …

# ietf-sr-policy-types (2)

- State:
  - binding-sid-oper-state
  - policy-admin-state
  - policy-oper-state
  - policy-down-reason
  - candidate-path-non-selection-reason
  - binding-sid-unavailable-reason

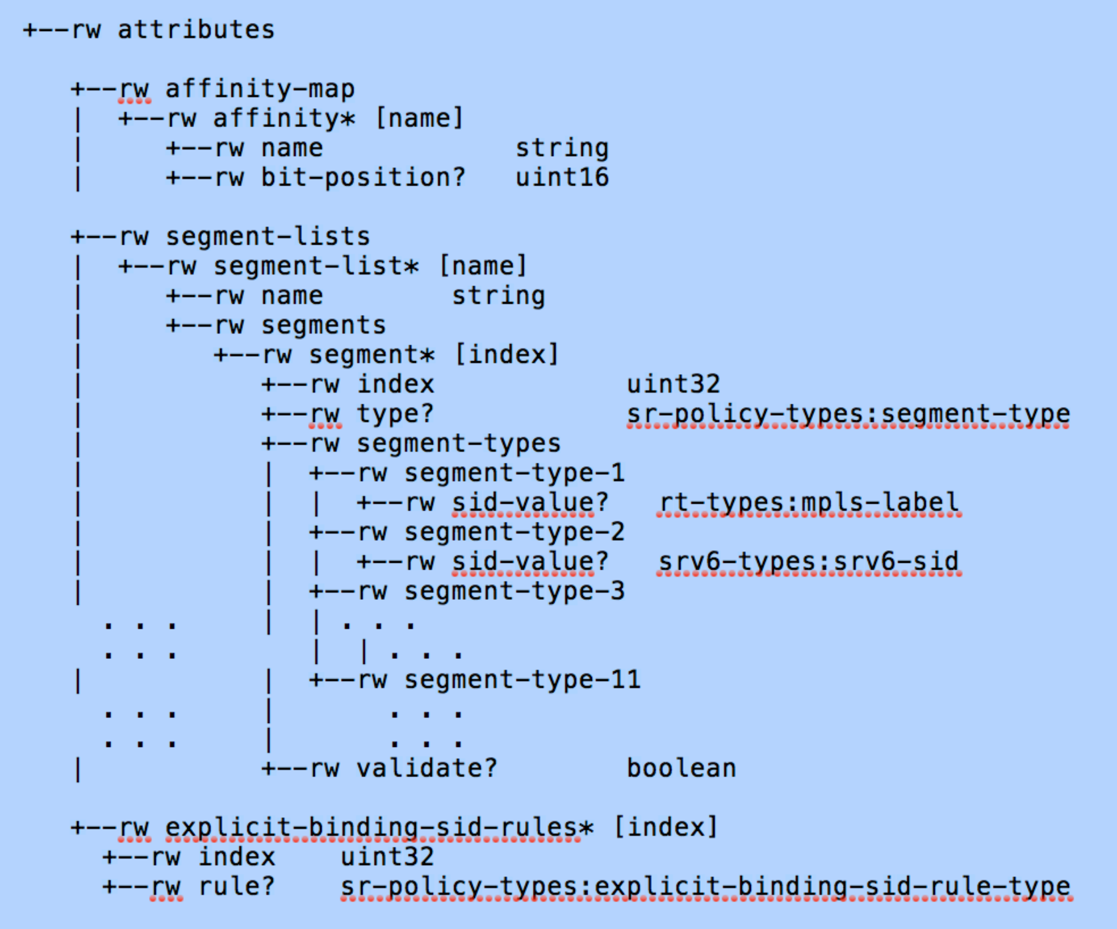
# ietf-sr-policy: Hierarchy



augments: /rt: routing

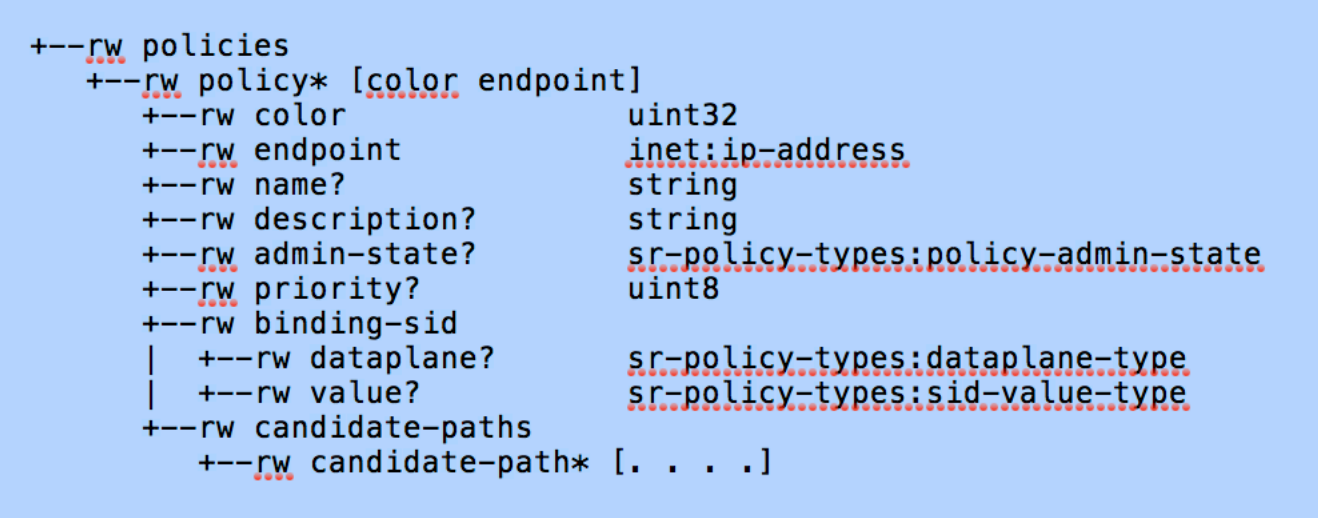
```
segment-routing
   traffic-engineering
      + attributes
      |   + affinity-map
      |   |  ....
      |   |
      |   + segment-lists
      |   |    segment-list* [name]
      |   |        segments
      |   |            segment* [index]
      |   |                ...
      |   + explicit-binding-sid-rules
      |        ...
      |
      + policies
           policy* [color endpoint]
              + ...
              |
              + binding-sid
              | ...
              |
              + candidate-paths
                  candidate-path* [protocol-origin originator discriminator]
                      + ...
                      |
                      + type
                          + explicit
                          |   segment-lists
                          |        segment-list* [ref]
                          |             ...
                          + dynamic
                              constraints
                                 ...
```

# ietf-sr-policy: Config - Attributes



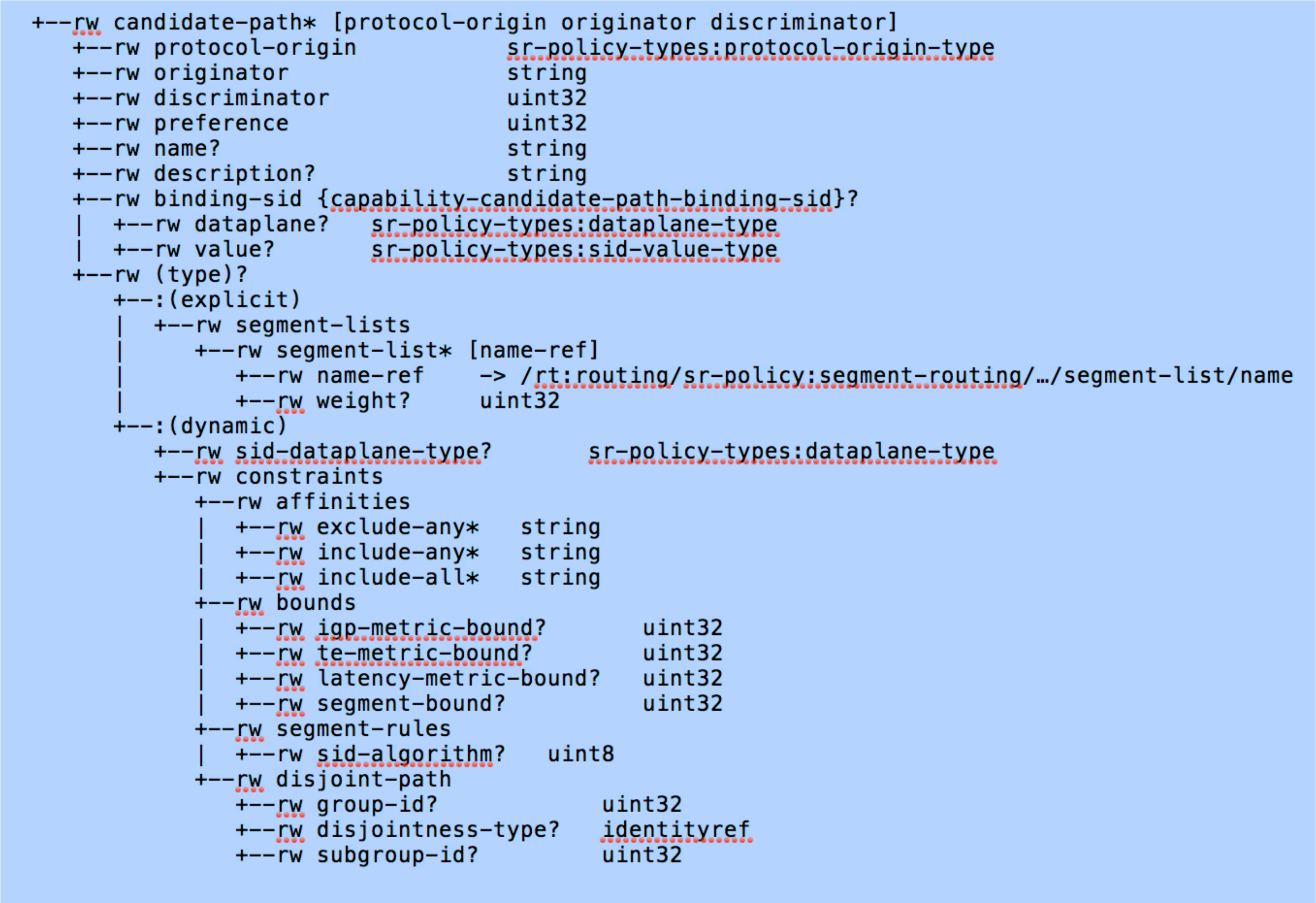
```
+--rw attributes

   +--rw affinity-map
   |  +--rw affinity* [name]
   |     +--rw name            string
   |     +--rw bit-position?   uint16

   +--rw segment-lists
   |  +--rw segment-list* [name]
   |     +--rw name        string
   |     +--rw segments
   |        +--rw segment* [index]
   |           +--rw index              uint32
   |           +--rw type?              sr-policy-types:segment-type
   |           +--rw segment-types
   |           |  +--rw segment-type-1
   |           |  |  +--rw sid-value?   rt-types:mpls-label
   |           |  +--rw segment-type-2
   |           |  |  +--rw sid-value?   srv6-types:srv6-sid
   |           |  +--rw segment-type-3
     . . .     |  | . . .
     . . .        |  | . . .
   |           |  +--rw segment-type-11
     . . .     |        . . .
     . . .     |        . . .
   |           +--rw validate?          boolean

   +--rw explicit-binding-sid-rules* [index]
     +--rw index     uint32
     +--rw rule?     sr-policy-types:explicit-binding-sid-rule-type
```

# ietf-sr-policy: Config - policy



```
+--rw policies
   +--rw policy* [color endpoint]
      +--rw color               uint32
      +--rw endpoint            inet:ip-address
      +--rw name?               string
      +--rw description?        string
      +--rw admin-state?        sr-policy-types:policy-admin-state
      +--rw priority?           uint8
      +--rw binding-sid
      |  +--rw dataplane?       sr-policy-types:dataplane-type
      |  +--rw value?           sr-policy-types:sid-value-type
      +--rw candidate-paths
         +--rw candidate-path* [. . . .]
```

# ietf-sr-policy: Config - candidate-path

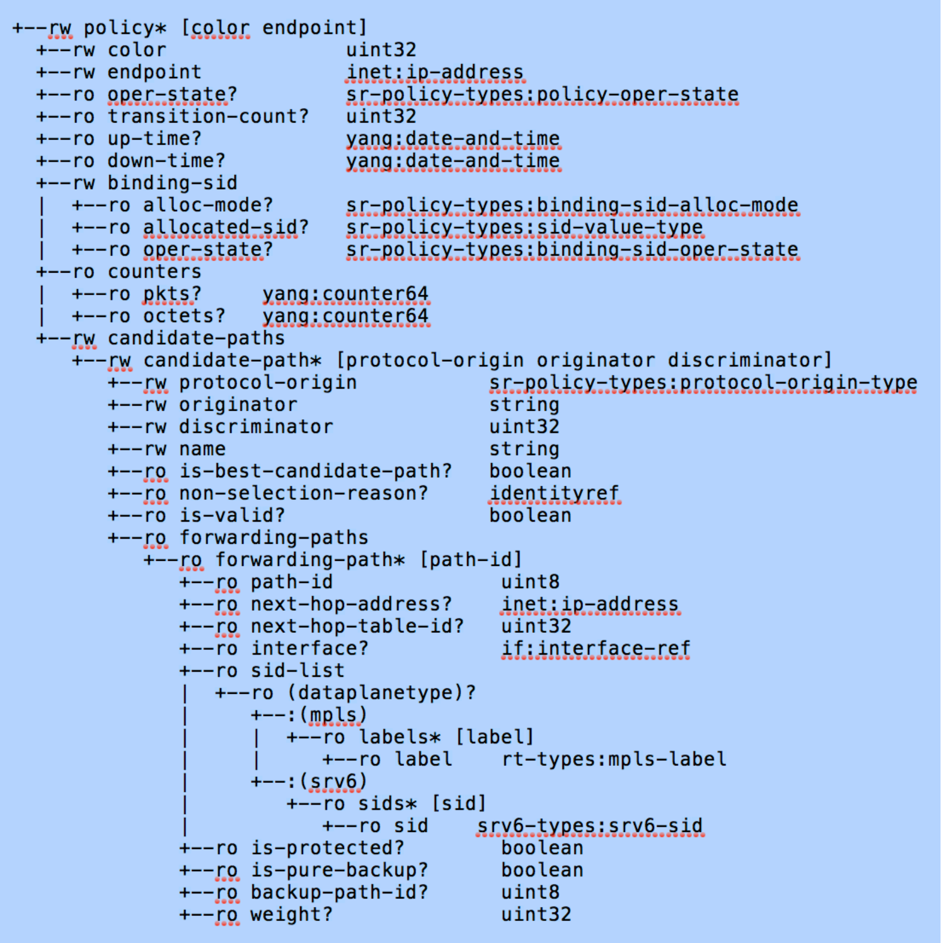


```
+--rw candidate-path* [protocol-origin originator discriminator]
   +--rw protocol-origin           sr-policy-types:protocol-origin-type
   +--rw originator                string
   +--rw discriminator             uint32
   +--rw preference                uint32
   +--rw name?                     string
   +--rw description?              string
   +--rw binding-sid {capability-candidate-path-binding-sid}?
   |  +--rw dataplane?   sr-policy-types:dataplane-type
   |  +--rw value?       sr-policy-types:sid-value-type
   +--rw (type)?
      +--:(explicit)
      |  +--rw segment-lists
      |     +--rw segment-list* [name-ref]
      |        +--rw name-ref    -> /rt:routing/sr-policy:segment-routing/…/segment-list/name
      |        +--rw weight?     uint32
      +--:(dynamic)
         +--rw sid-dataplane-type?       sr-policy-types:dataplane-type
         +--rw constraints
            +--rw affinities
            |  +--rw exclude-any*   string
            |  +--rw include-any*   string
            |  +--rw include-all*   string
            +--rw bounds
            |  +--rw igp-metric-bound?       uint32
            |  +--rw te-metric-bound?        uint32
            |  +--rw latency-metric-bound?   uint32
            |  +--rw segment-bound?          uint32
            +--rw segment-rules
            |  +--rw sid-algorithm?   uint8
            +--rw disjoint-path
               +--rw group-id?            uint32
               +--rw disjointness-type?   identityref
               +--rw subgroup-id?         uint32
```

# ietf-sr-policy: policy state

In addition to configured state

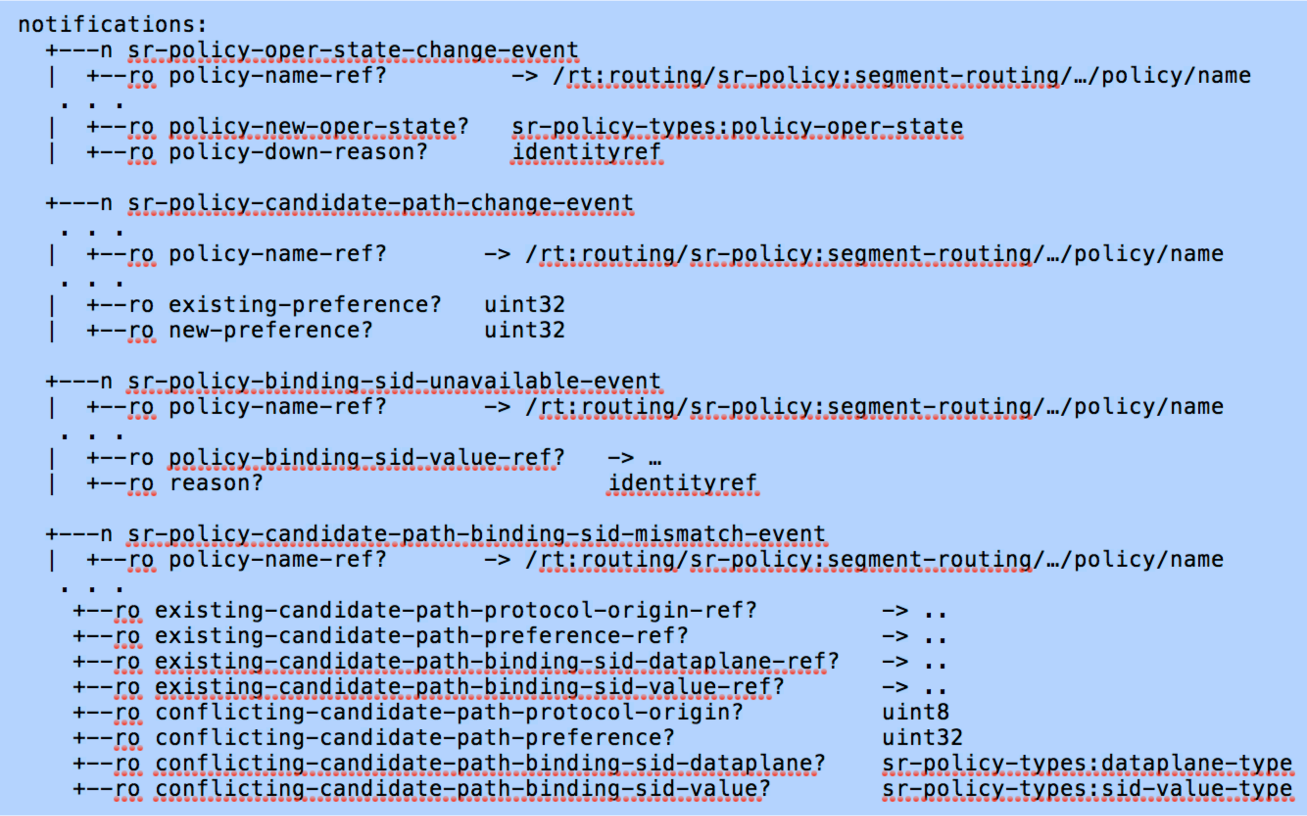
- policy operational state
- policy up/down timestamps
- policy BSID info such as alloc mode, actual value in-use,
- operational state, and forwarding stats
- Per candidate-path info such as:
    - Whether candidate-path is the best candidate-path
    - In case of non-best, the reason for such non-selection
    - Type of candidate-path - explicit or dynamic
    - Per segment-list information - such as validity of the segment-list, as well as forwarding state for a valid segment-list.



```
+--rw policy* [color endpoint]
  +--rw color               uint32
  +--rw endpoint            inet:ip-address
  +--ro oper-state?         sr-policy-types:policy-oper-state
  +--ro transition-count?   uint32
  +--ro up-time?            yang:date-and-time
  +--ro down-time?          yang:date-and-time
  +--rw binding-sid
  |  +--ro alloc-mode?      sr-policy-types:binding-sid-alloc-mode
  |  +--ro allocated-sid?   sr-policy-types:sid-value-type
  |  +--ro oper-state?      sr-policy-types:binding-sid-oper-state
  +--ro counters
  |  +--ro pkts?     yang:counter64
  |  +--ro octets?   yang:counter64
  +--rw candidate-paths
     +--rw candidate-path* [protocol-origin originator discriminator]
        +--rw protocol-origin           sr-policy-types:protocol-origin-type
        +--rw originator                string
        +--rw discriminator             uint32
        +--rw name                      string
        +--ro is-best-candidate-path?   boolean
        +--ro non-selection-reason?     identityref
        +--ro is-valid?                 boolean
        +--ro forwarding-paths
           +--ro forwarding-path* [path-id]
              +--ro path-id              uint8
              +--ro next-hop-address?    inet:ip-address
              +--ro next-hop-table-id?   uint32
              +--ro interface?           if:interface-ref
              +--ro sid-list
              |  +--ro (dataplanetype)?
              |     +--:(mpls)
              |     |  +--ro labels* [label]
              |     |     +--ro label    rt-types:mpls-label
              |     +--:(srv6)
              |        +--ro sids* [sid]
              |           +--ro sid    srv6-types:srv6-sid
              +--ro is-protected?        boolean
              +--ro is-pure-backup?      boolean
              +--ro backup-path-id?      uint8
              +--ro weight?              uint32
```

# ietf-sr-policy: notifications

- policy status: policy operational state changes
- Candidate-path active status and changes
- Explicit Binding SID collision/unavailability events



```
notifications:
  +---n sr-policy-oper-state-change-event
  |  +--ro policy-name-ref?         -> /rt:routing/sr-policy:segment-routing/…/policy/name
   . . .
  |  +--ro policy-new-oper-state?   sr-policy-types:policy-oper-state
  |  +--ro policy-down-reason?      identityref

  +---n sr-policy-candidate-path-change-event
   . . .
  |  +--ro policy-name-ref?       -> /rt:routing/sr-policy:segment-routing/…/policy/name
   . . .
  |  +--ro existing-preference?   uint32
  |  +--ro new-preference?        uint32

  +---n sr-policy-binding-sid-unavailable-event
  |  +--ro policy-name-ref?       -> /rt:routing/sr-policy:segment-routing/…/policy/name
   . . .
  |  +--ro policy-binding-sid-value-ref?    -> …
  |  +--ro reason?                          identityref

  +---n sr-policy-candidate-path-binding-sid-mismatch-event
  |  +--ro policy-name-ref?       -> /rt:routing/sr-policy:segment-routing/…/policy/name
   . . .
    +--ro existing-candidate-path-protocol-origin-ref?         -> ..
    +--ro existing-candidate-path-preference-ref?               -> ..
    +--ro existing-candidate-path-binding-sid-dataplane-ref?   -> ..
    +--ro existing-candidate-path-binding-sid-value-ref?       -> ..
    +--ro conflicting-candidate-path-protocol-origin?          uint8
    +--ro conflicting-candidate-path-preference?               uint32
    +--ro conflicting-candidate-path-binding-sid-dataplane?    sr-policy-types:dataplane-type
    +--ro conflicting-candidate-path-binding-sid-value?        sr-policy-types:sid-value-type
```

# Pending Items

- Configuration and Specification of:
  - Traffic steering over SR policy
  - ODN templates
- Executables (RPC actions)

# Next Steps

- Asking for WG adoption
- Further updates as per WG feedback