

VEGVÍSIR
systems

Auto-bandwidth for Segment Routing

Dmytro Shypovalov
dmytro@vegvisir.ie

Agenda

- Legacy MPLS vs Segment Routing
- SR-TE overview
- SR auto-bandwidth design with telemetry sampler
- SR-TE mesh templates and ACTN
- Implementation example
- Future design

RSVP-TE auto-bandwidth

- RSVP can signal bandwidth reservations per LSP
- Each router measures traffic rate on its LSP and adjusts bandwidth reservations accordingly

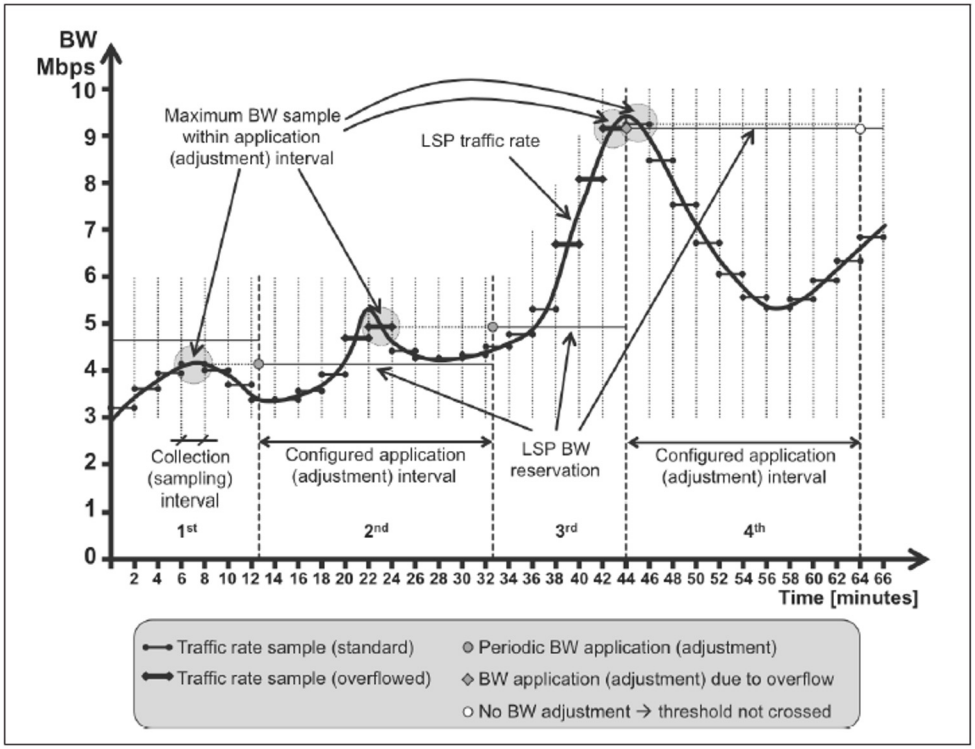


Image source: MPLS in the SDN era, Figure 14-3

Problems with RSVP-TE

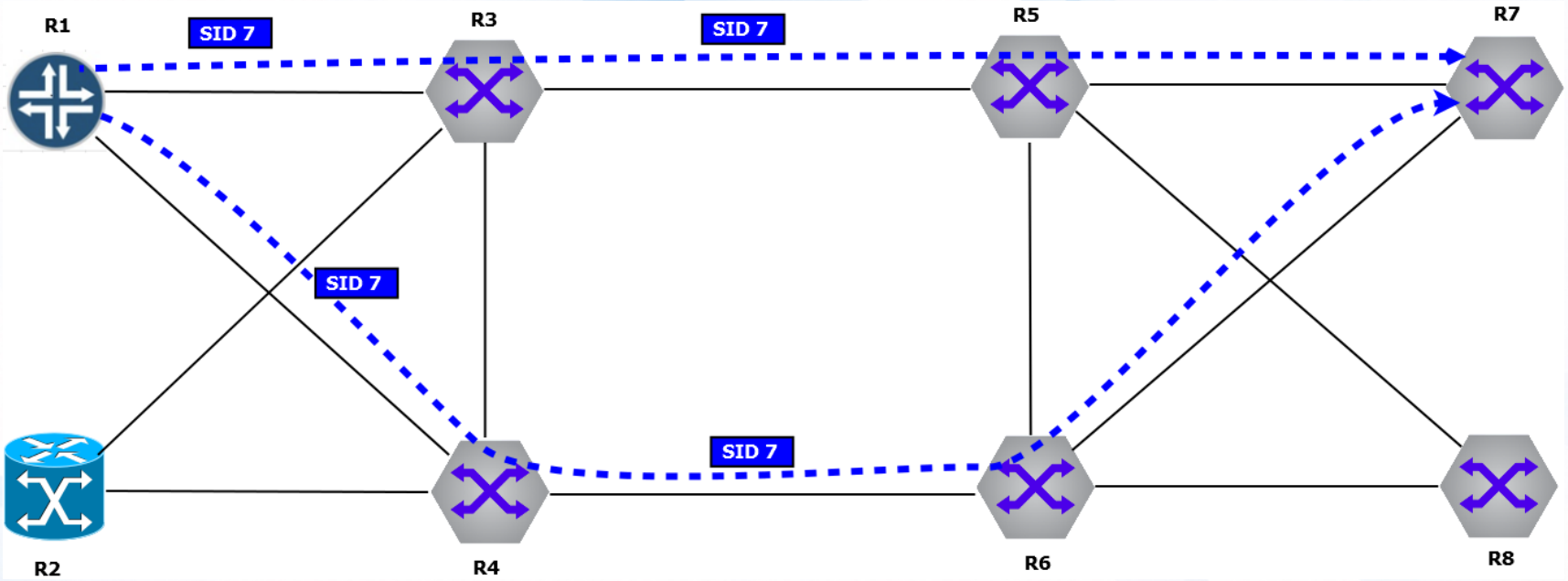
- Operational complexity, poor scalability, poor interoperability
- No deterministic routing, each reoptimization can yield different results
- Slow reaction on traffic bursts (some improvement with the overflow feature)
- Auto-bandwidth can't adequately react on changing BGP destinations
- RSVP has no mechanism to map different services on the same PE to different LSP
- Adding new routers requires configuring new LSP on all existing routers

Segment Routing

- SR is simpler, scales better and has better multi-vendor interop than legacy MPLS
- No need for an extra protocol (LDP/RSVP)
- Most optimal fast convergence with TI-LFA
- Flex Algo for basic traffic engineering and network slicing
- SR dataplane options: MPLS or SRv6
  - MPLS is more widely supported now, but the industry is moving towards SRv6

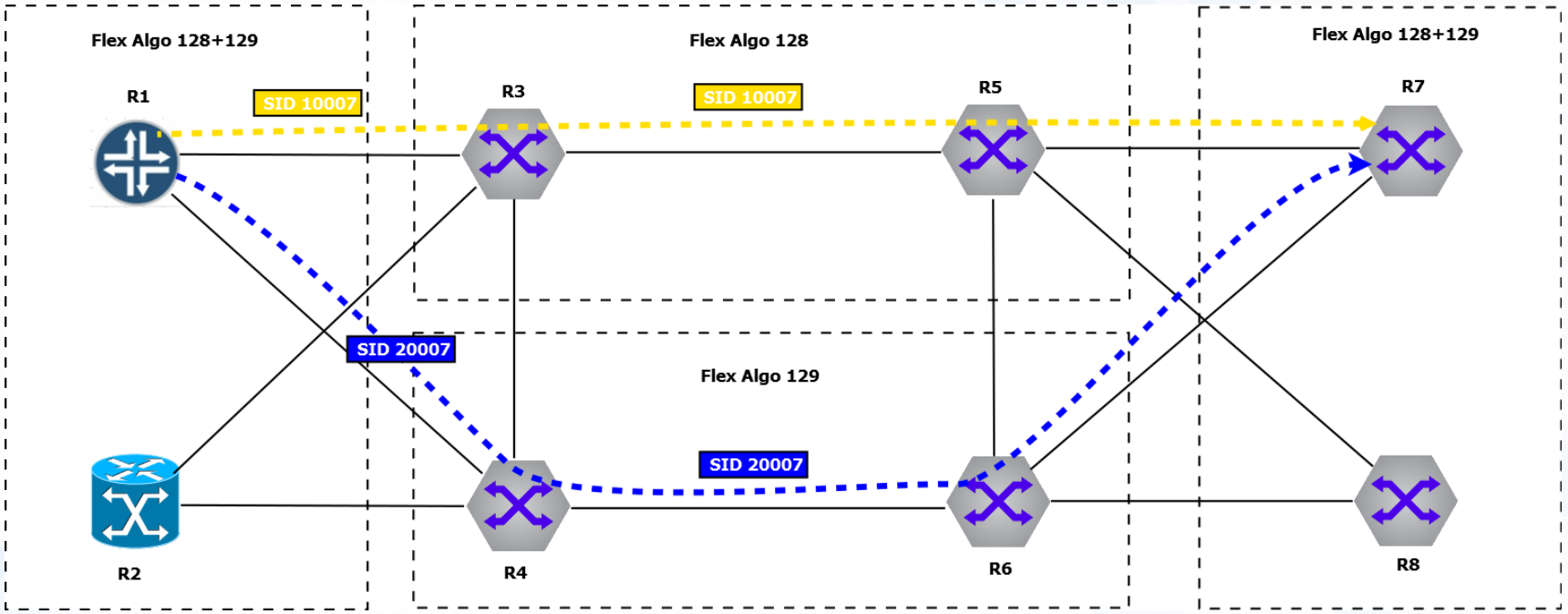
Basic SR

- Normally SR forwarding follows the shortest path, including ECMP when applicable



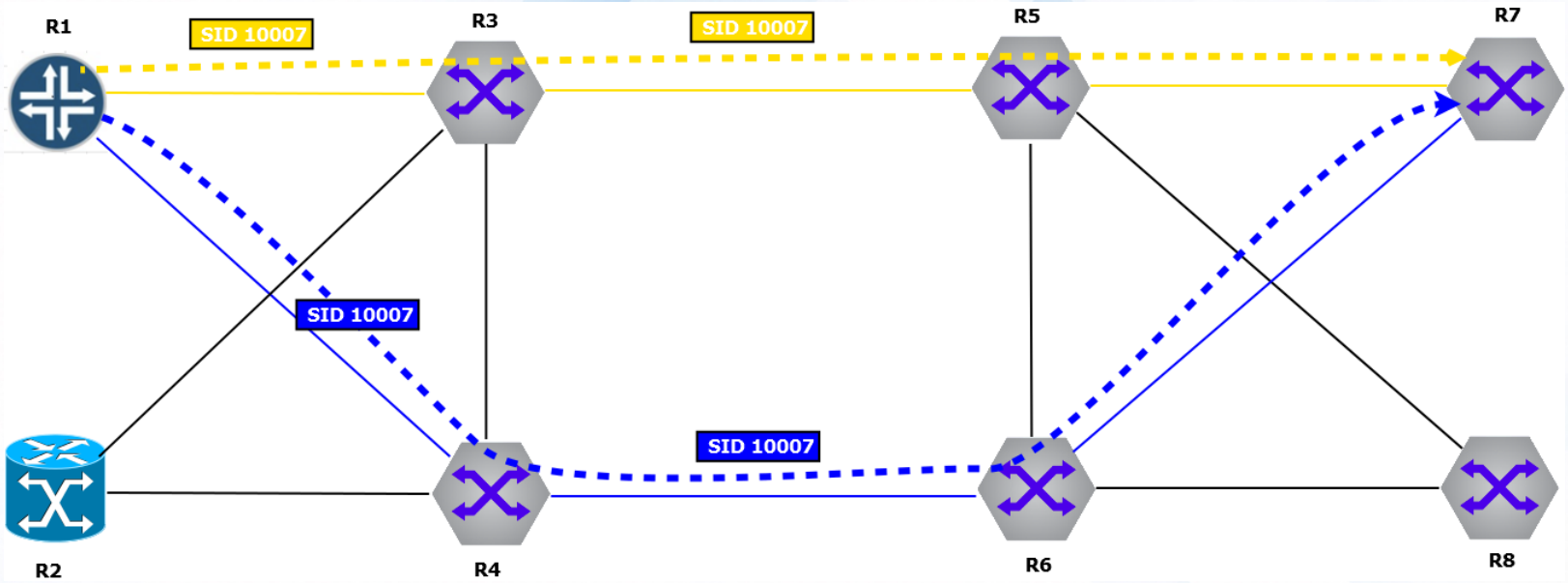
SR: Flex Algo

- Flex Algo enables basic TE capabilities with simple config
- E.g. flex algo for low latency traffic, or basic path diversity



SR-TE

- SR-TE can manipulate traffic in every possible way; use constraints like bandwidth, delay, admin groups etc
- E.g. provision LSP1 with YELLOW affinity, LSP2 with BLUE affinity



SR-TE vs RSVP-TE

- Better scalability and interoperability
- Standardized method of mapping traffic to an LSP using color
- Tactical TE: most traffic follows the IGP shortest path, deploy some TE policies only when required
- SR LSP are stateless, routers are unaware of transit LSP
  - Statelessness enables better scalability
  - However, a controller (PCE) is required for bandwidth reservations
  - For auto-bandwidth, some feedback loop between a controller and a PCE is required

When to use PCE

Features

- Bandwidth reservations
- Disjoint paths
- Egress Peer Engineering
- Full mesh TE or close

Equipment

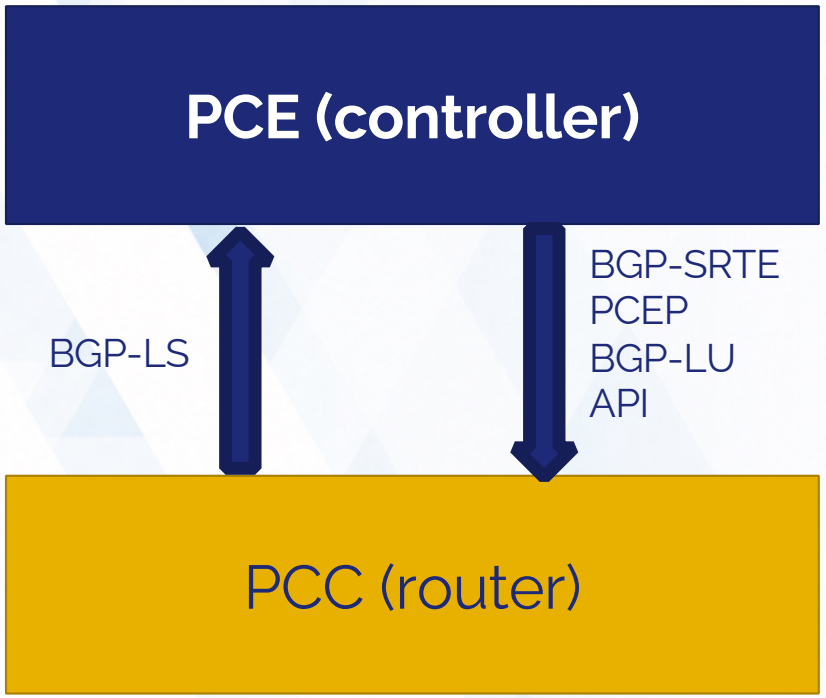
- Multi-vendor network*
- Less mature SR implementations
- Frequent new deployments

Automation

- Highly automated network
- Use of telemetry, analytics

SR PCE basics

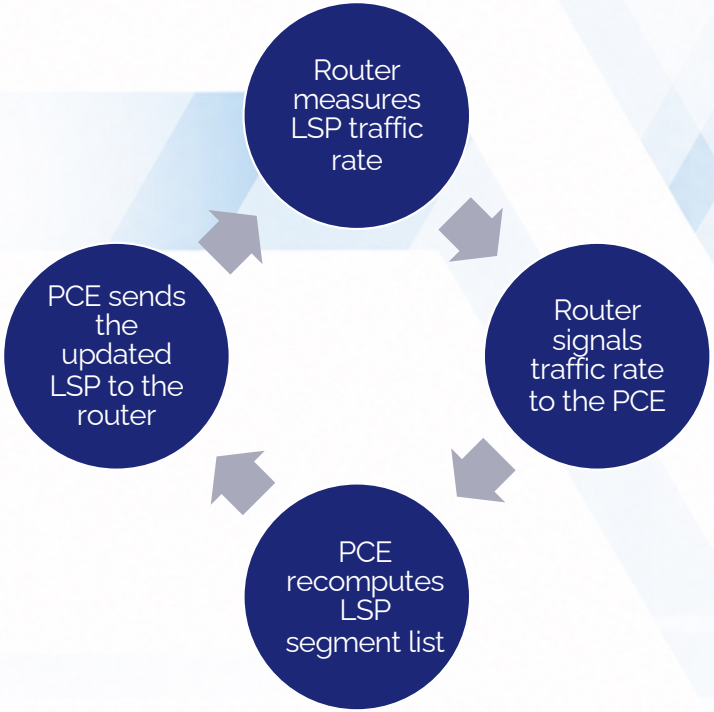
- PCE receives topology via BGP-LS
- PCE computes policies
- PCE sends policies to the network using either of the following:
  - **BGP-SRTE** (best protocol, use this one!)
  - **PCEP** (please don't use it)
  - **BGP-LU** (for basic or legacy routers)
  - **API** (vendor-specific)



BGP-LS route types

- Type 1 – Node NLRI (system ID, SRGB…)
- Type 2 – Link NLRI (IP, TE constraints)
- Type 3 – IPv4 prefix (prefix, SID)
- Type 4 – IPv6 prefix (prefix, SID)
- Type 5 – SR policy

Auto-bandwidth feedback loop



Feedback loop implementations

RFC8733

- PCEP for LSP traffic rate signaling
- PCEP for LSP provisioning

RFC9857

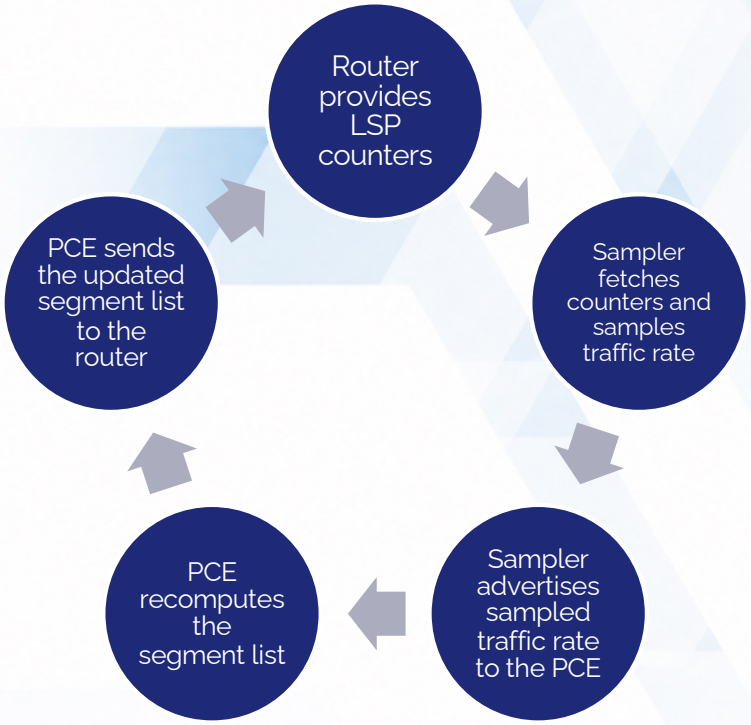
- Advertise LSP traffic rate with BGP-LS SR policy route + bandwidth constraint TLV
- Use any protocol to provision LSP

API

- Use API to advertise LSP traffic rate
- Use any protocol to provision LSP
- Vendor-specific

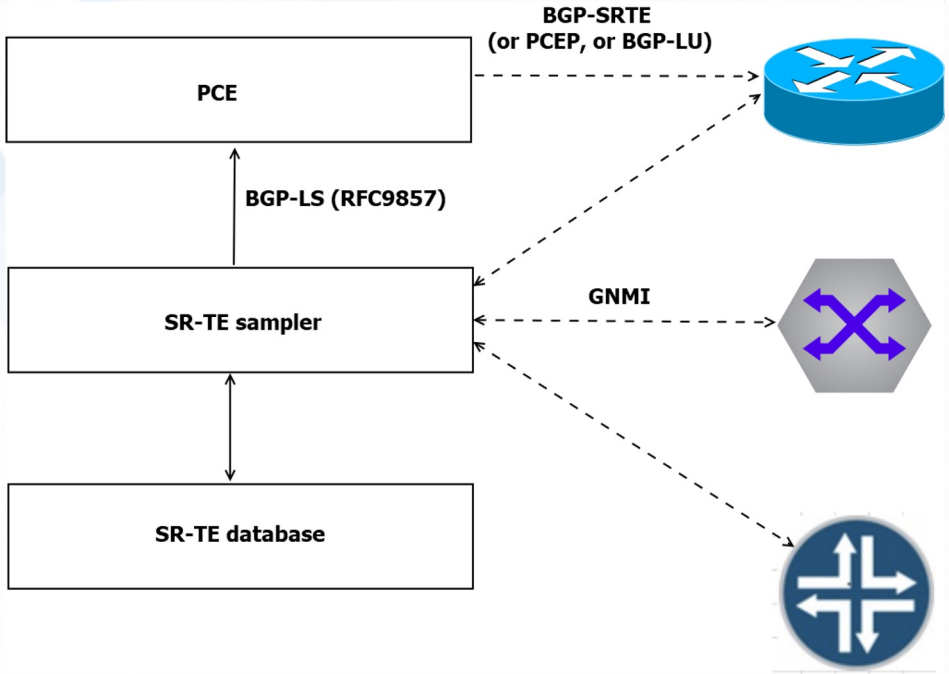
- The problem with all these – the router has to do the sampling
  - **Most routers support SR LSP counters, but don't support any of the above**

Auto-bandwidth feedback loop – with sampler



SR-TE bandwidth sampler

- Bandwidth sampler collects SR LSP counters via GNMI
- Based on adjust interval, the sampler advertises the LSP traffic rate to the PCE using BGP-LS type 5 route
- Currently works with IOS-XR, JUNOS, EOS
- The sampler is open source: https://github.com/Vegvisir-Systems/srte-sampler/



Sampler - example

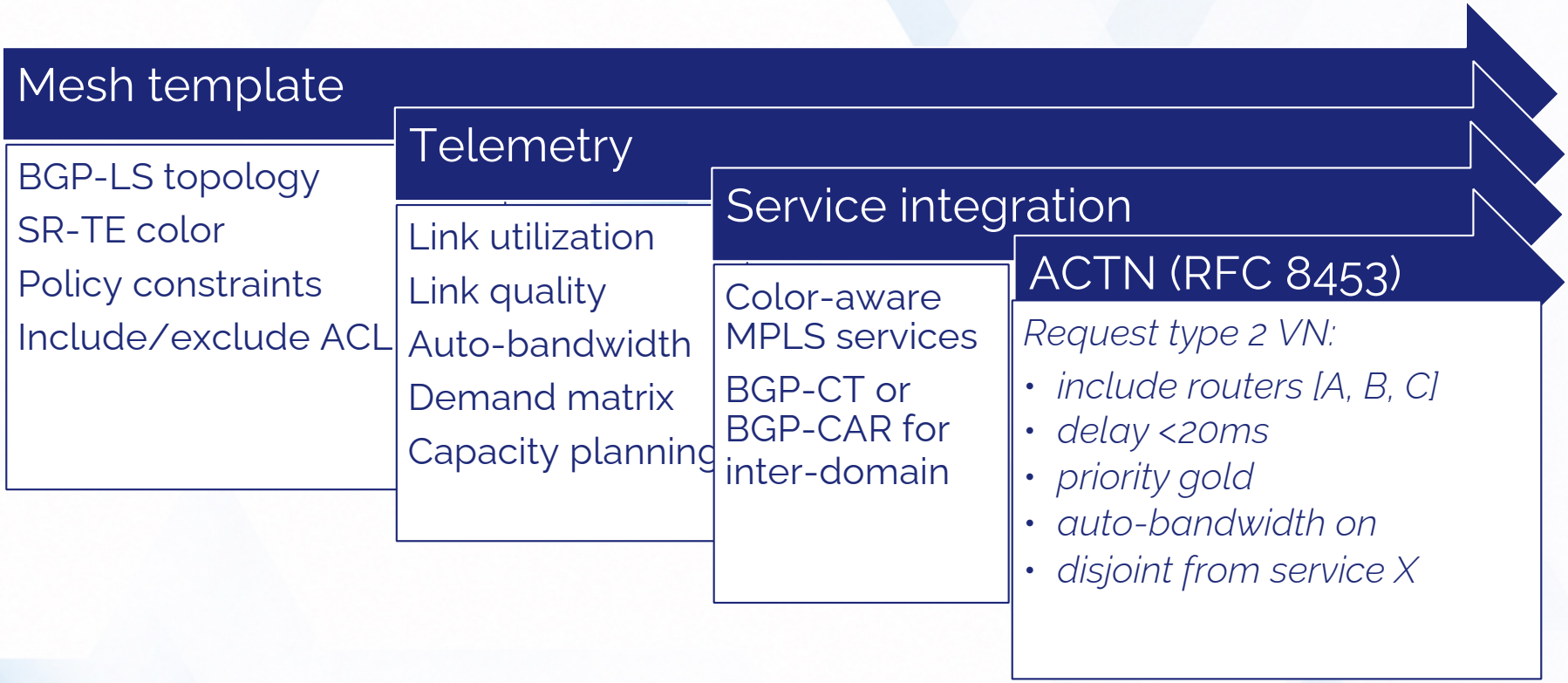
```
lmk-vm103-dev-bw-sampler#show sampling policies [1.1.1.1][2.2.2.2][101]
Detailed sampling policies information
Number of policies: 1, active 1, stale 0

Sampled policy entry for [1.1.1.1][2.2.2.2][101]
  Router-id: 1.1.1.1
  Endpoint: 2.2.2.2
  Color: 101
  Rate 39.924 Gbps, calculated from 6 samples within 50.4 seconds
  Last updated: 0:00:49 ago
```

Auto-bandwidth with sampler

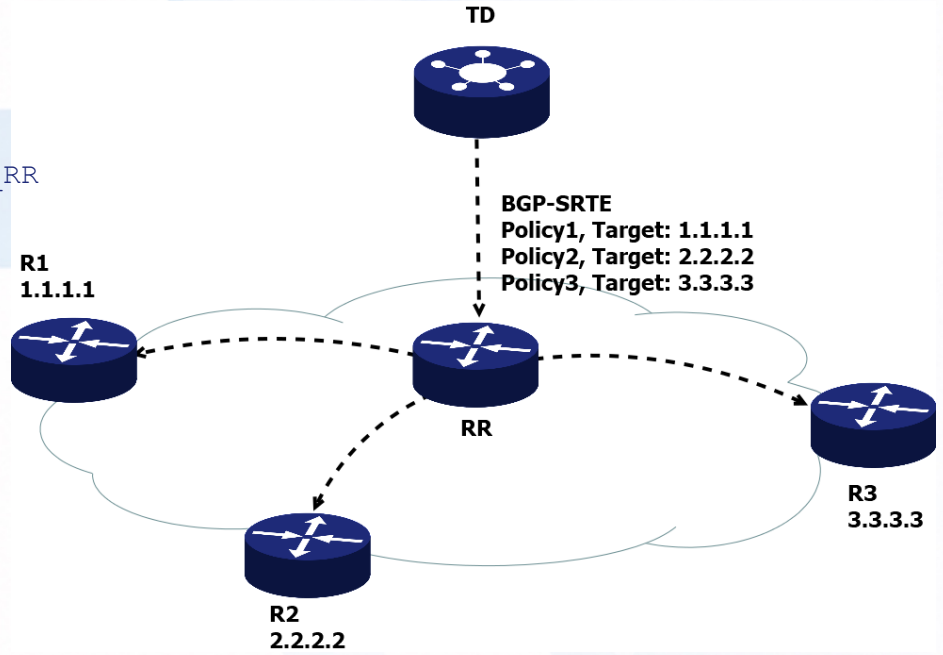
- Sampler advertises BGP-LS type 5 routes with traffic rate to the PCE
- PCE adjusts bandwidth reservations and updates segment lists
- But is this good enough?
- We just replicated the behavior of RSVP-TE auto-bandwidth
  - Maybe with better scalability, and deterministic results

SR-TE mesh templates – foundation for intent-based networking



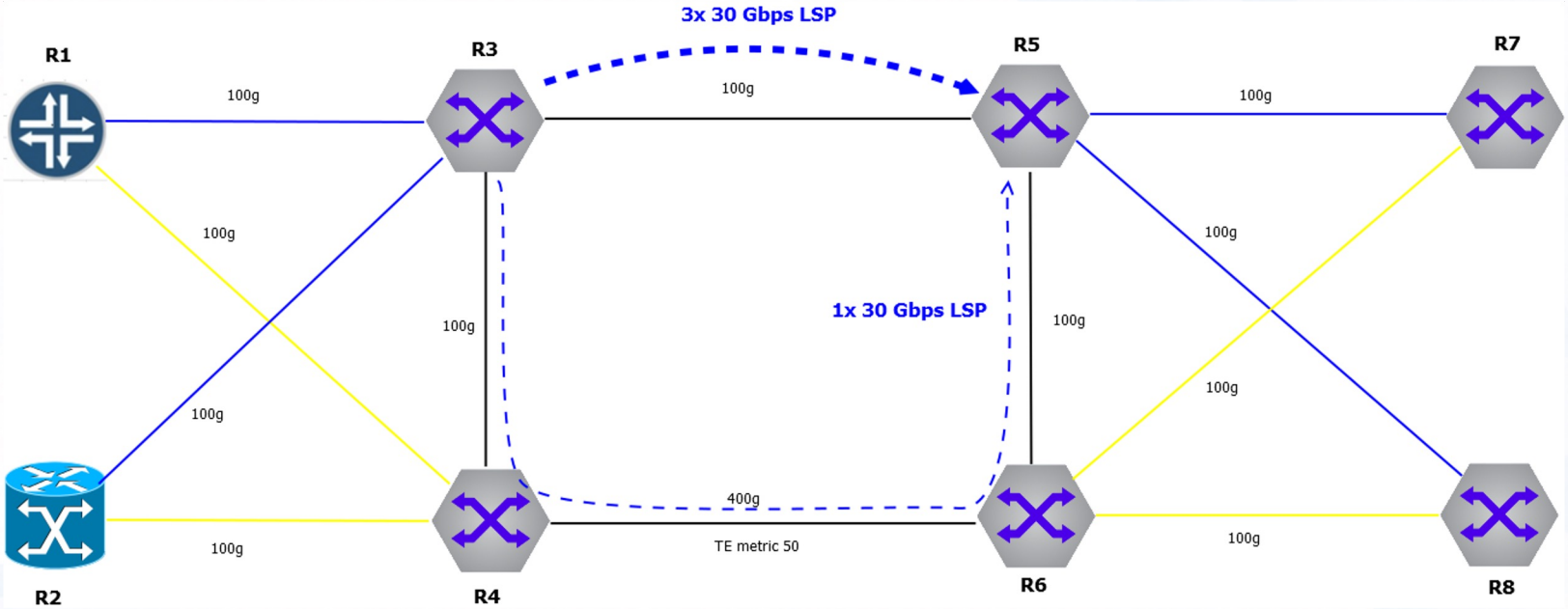
SR-TE mesh templates - example

```
traffic-eng mesh-templates
   !
   template TOPO_101_BLUE
      topology-id 101
      color 11101
      install indirect srte peer-group TOPO_101_RR
      !
      candidate-path preference 100
         affinity-set CORE_LINKS
```



SR-TE mesh templates + auto-bandwidth

- We need an LSP mesh between PE (R1, R2, R7, R8)
- Core link R3-R5 has lower latency than R4-R6, however it has limited capacity
- Each LSP sends ~30 Gbps



Mesh template + auto-bw: TD config

```
traffic-eng capacity-profiles
   !
   profile AUTO_BW
      bandwidth auto
!
traffic-eng mesh-templates
   !
   template TOPO101_BLUE_AUTOBW
      topology-id 101
      color 101
      access-list ipv4 ALL_PE_IPV4
      install indirect srte peer-group RR
      !
      candidate-path preference 100
         metric te
         affinity-set BLUE_ONLY
         capacity-profile AUTO_BW
```

Mesh template + auto-bw: outputs

```
lmk-vm103-dev-bw-sampler#show sampling policies
Sampling policies information
Number of policies: 12, active 12, stale 0
Status codes: ~ stale

    Policy                                                  Rate                  Last updated
    ---snip---
    [1.1.1.1][2.2.2.2][101]                          30.326 Gbps                        0:00:08
    [1.1.1.1][7.7.7.7][101]                          29.849 Gbps                        0:00:08
    [1.1.1.1][8.8.8.8][101]                          31.015 Gbps                        0:00:08
```

```
lmk-vm102-dev-td1#show traffic-eng mesh-template policies
Traffic-eng policy information per mesh-template
-----
Mesh-template: TOPO101_BLUE_AUTOBW
Status codes: * valid, > active, s - admin down
       Policy name                          Headend     Endpoint    Color      Reserved bandwidth
       ---snip---
    *>  mesh_h1.1.1.1_e8.8.8.8_c101        1.1.1.1     8.8.8.8     101        31.015 Gbps(auto)
    *>  mesh_h1.1.1.1_e7.7.7.7_c101        1.1.1.1     7.7.7.7     101        29.849 Gbps(auto)
    *>  mesh_h1.1.1.1_e2.2.2.2_c101        1.1.1.1     2.2.2.2     101        30.326 Gbps(auto)
```

Redundancy and failover

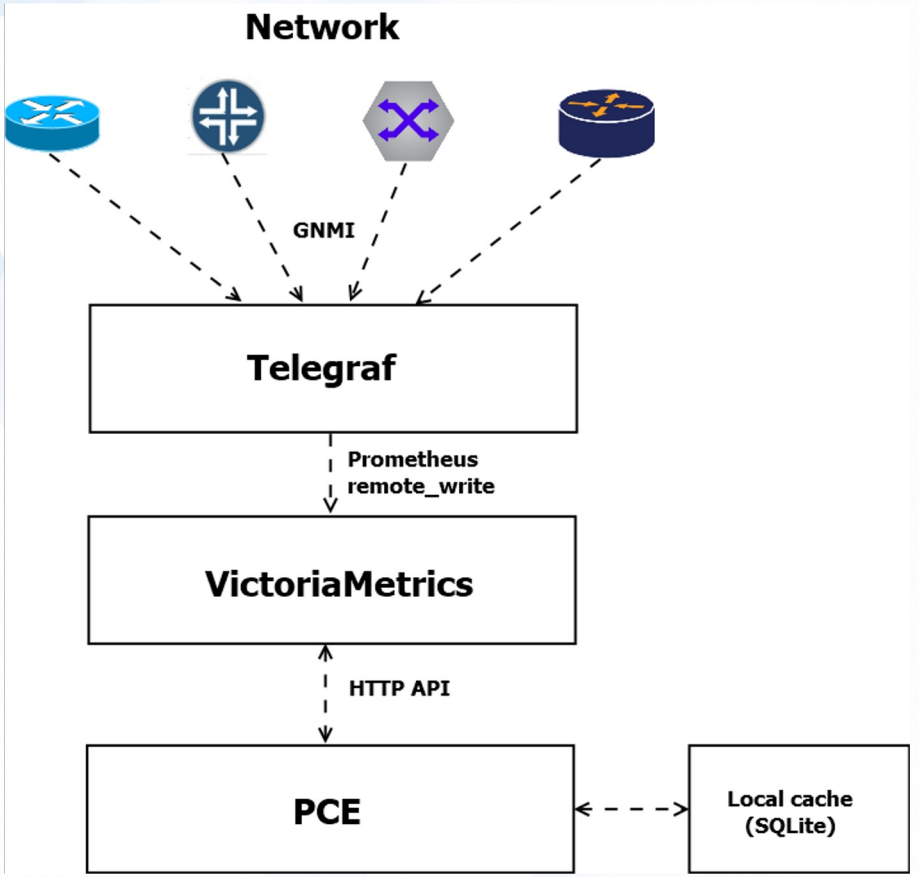
- **The solution is based on open-standards (BGP-LS, BGP-SRTE), hence it's robust, easy to understand, and fails in predictable ways**
- **SR-TE sampler redundancy is not required – if fails, just restart on another machine**
  - If routers support LSP traffic rate advertisement – sampler not required
- **PCE - deploy as many as you want, no proprietary state-sync is needed (think of a route reflector)**
  - Each PCE is configured with different SRTE distinguisher, so each router receives multiple copies of the same SR-TE policy NLRI from different PCE
  - If one is withdrawn, router just uses another one
  - Catastrophic failure of all PCE: traffic just follows the shortest path

Problems with sampled bandwidth

- **Advertising traffic rate (rather than raw counters) to the PCE is easy to implement, but not ideal**
  - No historical data, trends, predictions
  - Sampling method not controlled by PCE (e.g. use 98% percentile)
  - No SID/locator counters and interface counters
  - No link quality and reliability data
- **A better idea would be a separate telemetry layer, from which PCE can get relevant data**
- **Most ISP already have network monitoring systems that use SNMP or GNMI/OpenConfig**

Auto-bandwidth: future design

- Use Telegraf and VictoriaMetrics to collect raw counters
  - Or any existing system the ISP already has
- PCE queries raw counters and samples traffic rate
- PCE can correlate LSP counters with SID and interface counters
- PCE has a proper centralized view of the entire traffic matrix
- Can freeze/rollback auto-bw
- Can run analytics/predictions
- More flexible and resilient design



Auto-bandwidth: conclusion

- Implementing auto-bw with SR-TE is possible today, and is quite easy
- Getting it right is not that easy. The bare minimum for production:
  - Must reliably work across all vendors
  - Must have a clearly understood and deterministic sampling method
  - Different controllers must produce the same result (ideally without requiring state sync)
- A more advanced design that is actually better than RSVP-TE:
  - Quick reaction on traffic bursts and changing BGP destinations
  - Centralized view of traffic matrix (including IGP shortest path traffic)
  - Analytics, predictions, digital twin simulations
  - State freeze/rollback for troubleshooting
- Hopefully this will run in production some time in 2027