# P2MP Transport Using Chain Replication in Segment Routing

draft-shen-spring-p2mp-transport-chain-02

Yimin Shen, Jeffrey Zhang - Juniper Networks

Rishabh Parekh - Cisco Systems

Hooman Bidgoli - Nokia

Yuji Kamite - NTT Communications

IETF 108, July 2020

Motivation – Stateless P2MP Transport

ingress replication

- ✓ Single-point provisioning at root
- ✓ Stateless
- ○ P2P tunnels, one per leaf
- ○ No traffic optimization
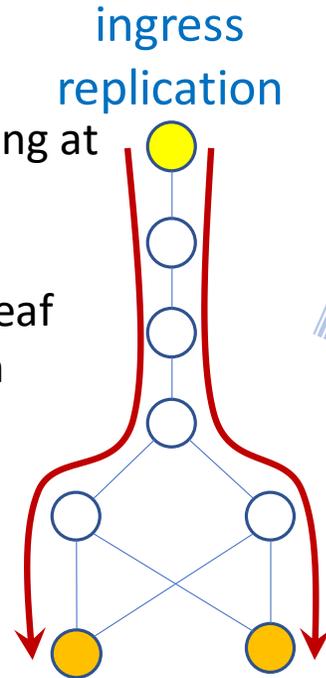
stateless P2MP chains

controller-driven P2MP tree
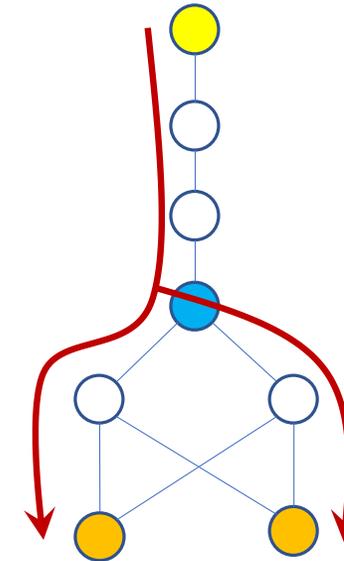
- ✓ P2MP tree
- ✓ Max traffic optimization
- ○ Controller-provisioned branch nodes
- ○ Stateful, with a set of replication SIDs on each branch node

- ○ P2MP chains
- ✓ Traffic optimization
- ✓ Stateless
- ✓ Single-point provisioning at root

P2MP Transport Using P2MP Chains

- A P2MP chain is a single-path tunnel that reaches multiple leaf nodes along the path.

- Root sends packets over a small set of P2MP chains.

- Applicable to all topologies. Benefits the most for some types of topology.
  - Ring topology: One P2MP chain per multicast stream.
  - Linear topology: Reduced traffic and tunnels across domains.
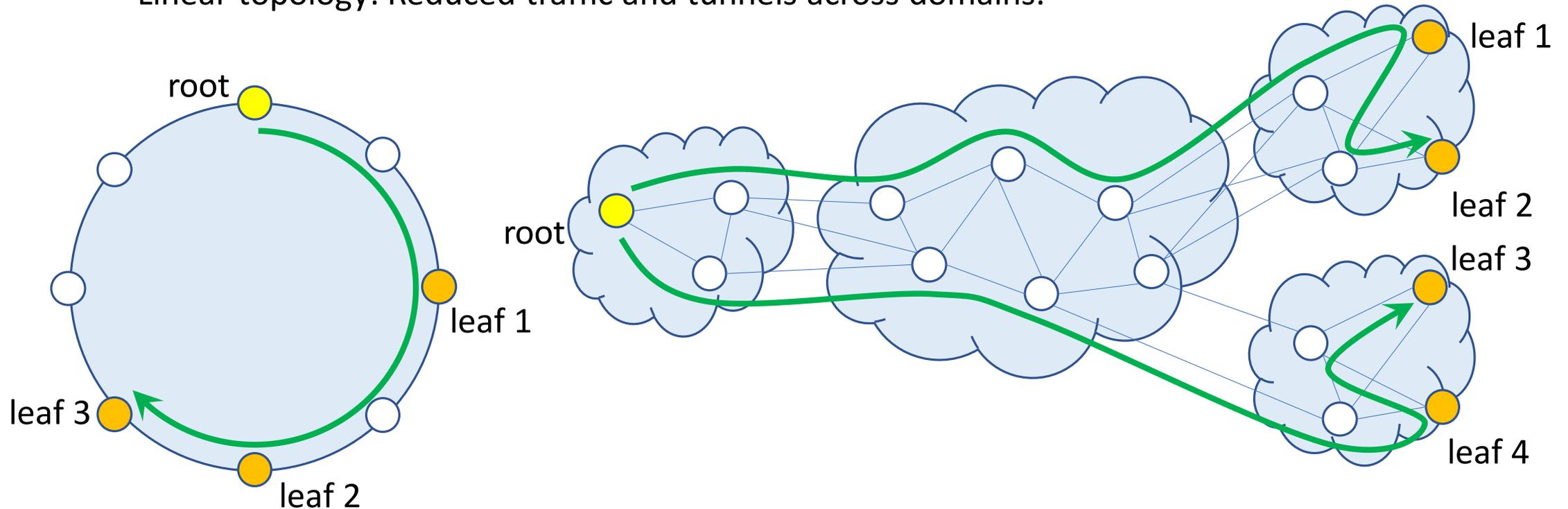
P2MP Chain - Chain Replication and Bud Segments

- Tail-end leaf node is a normal receiver.
- One or more transit leaf nodes, each acting as a **bud node**, i.e. both transit and receiver.
  - Modeled as **bud segments,** with **bud-SIDs**.
- A P2MP chain comprises a list of SIDs with transit leaf nodes represented by bud-SIDs.

transit leaf

tail-end leaf

R → T1 → L1 → T2 → L2

forwarding   forwarding   chain replication   forwarding   forwarding

local processing   local processing
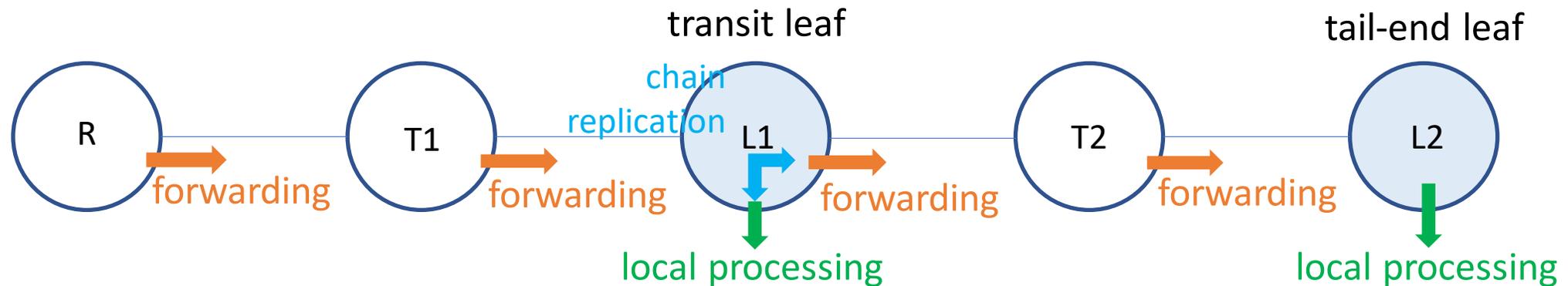
bud segment : replicate P → P1
forward P along a chain
process P1 locally

Example

A1 – A7: adj-SIDs
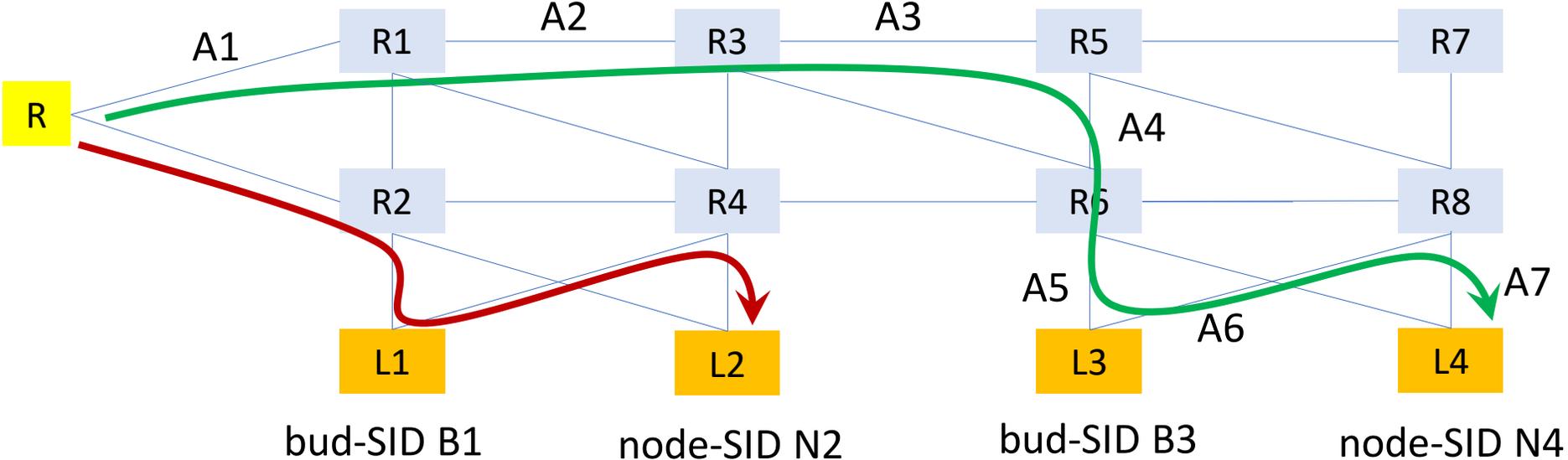
A multicast stream to L1 ~ L4, using two chains:

- Red Chain to L1 and L2 takes the shortest path from R to L1, and from L1 to L2.
  - SID list = {B1, N2}.
- Green Chain to L3 and L4 takes an explicit path from R to L3, and from L3 to L4.
  - SID list = {A1, A2, A3, A4, A5, B3, A6, A7}

Bud Segments

- Nodal segments
  - Two per node: one for SR-MPLS, and one for SRv6
- Global segments
  - Bud-SIDs are allocated from SRGB
- Routable segments, via shortest paths
  - Can also be used with adj-SIDs to construct explicit paths
- Advertised by ISIS/OSPF/BGP
  - *Similar to node segments*
- Building blocks of stateless P2MP, shared by all multicast streams
- Other use cases
  - Traffic mirroring, traffic monitoring, etc.

General Forwarding Flow of Bud Segment

Incoming packet P
Active SID == bud-SID

Replicate P → P1

P

P1

Perform Next on bud-SID;
Forward P based on next SID

Perform a sequence of Next on all the SIDs of P2MP chain;
Process P1 locally
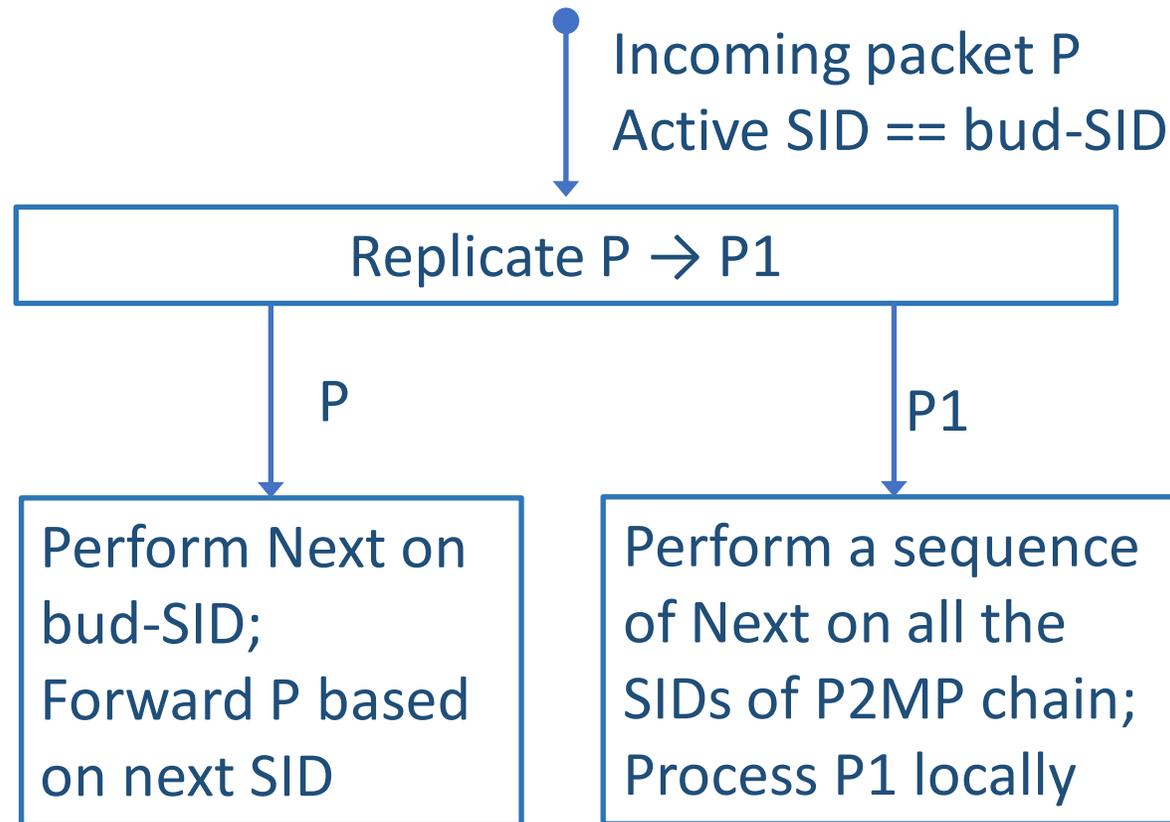
Forwarding Flow in SR-MPLS

To process P1 locally, must pop all the labels of P2MP chain.

- If there is a service label, root inserts an "End-of-Chain" (EoC) label between service label and P2MP chain labels.

- EoC is a new Extended Special-Purpose Label, as <XL = 15, EoC>.
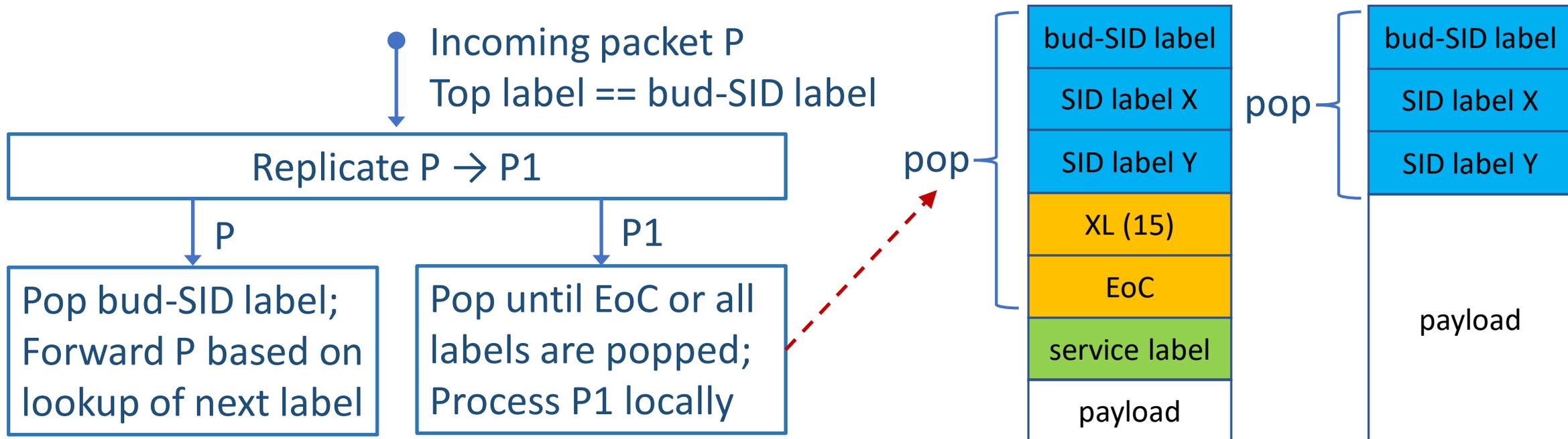
Forwarding Flow in SRv6

Packet: IPv6 header + SRH (P2MP chain SIDs) + IP/L2 header + …

Incoming packet P
DA == bud-SID

Replicate P → P1

P

P1

Adjust SRH;
Forward P based
on next SID

Remove outer IPv6
header and SRH;
Process P1 locally
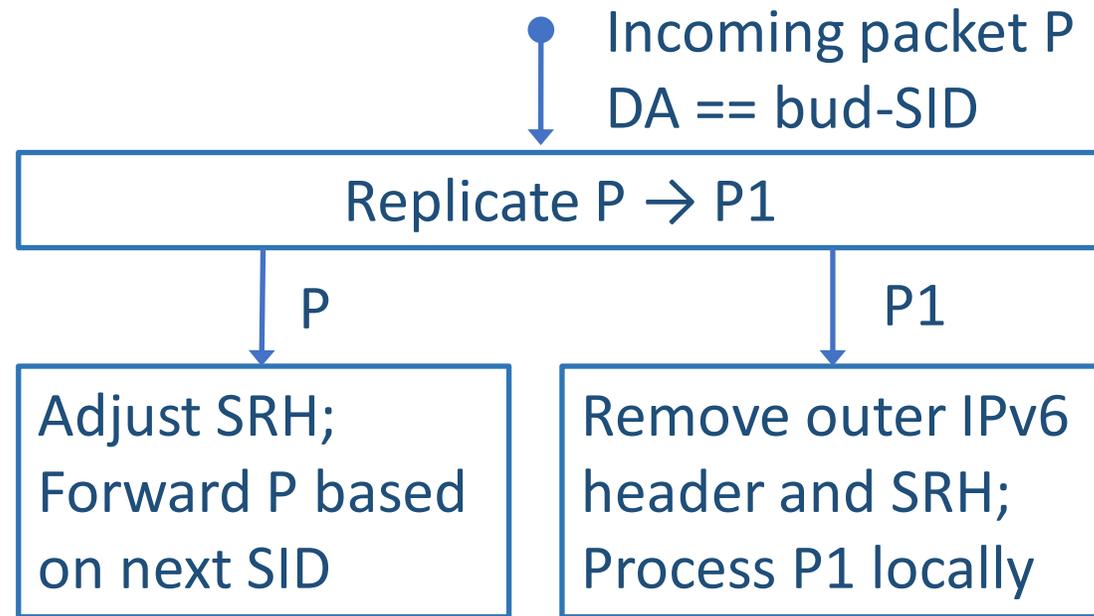
Path Computation

- P2MP chain computation can be supported by extending P2P path computation algorithms.
  - Max hop count per chain
    - Maximum delay allowed for a packet to accumulate before reaching a tail-end leaf node.
  - Max length of SID list
    - Max number of SIDs that a root node may apply to a packet.
    - Typically a limit of forwarding hardware.
  - Max number of leaf nodes per chain
  - Max number of hops between consecutive leaf nodes
  - Max number of times a link or node can be traversed by a chain
- Leaf groups
  - A leaf group is an ordered list of leaf nodes as loose hops.
  - Each chain covers the leaf nodes of a given group.

Next Steps

- Welcome review and comments from SPRING WG and MPLS WG.

- Protocol extensions for bud segment advertisement.

- Request an early allocation of *End-of-Chain Label* from the Extended Special Purpose Label Registry by IANA – *after WG adoption; for early implementation.*

Thank You