SEMANA DE CAPACITAÇÃO

Parceria

JUNIPER NETWORKS | Engineering Simplicity

Realização

ceptro.br nic.br

EVOLUÇÃO DA TECNOLOGIA DE TRANSPORTE LAYER 2 ETHERNET VPN (EVPN)

Eduardo Haro – eharo@juniper.net

Systems Engineer

Juniper Networks | Engineering Simplicity

AGENDA

- Introduction and drivers for EVPN
- EVPN architectural model and building blocks
- EVPN service types
- EVPN route types
- EVPN operations
- EVPN multi-homing
- EVPN IP Routing (Type 5)
- BUM optimization – ARP flooding
- BUM optimization – multicast flooding
- DC architectures using EVPN

Introduction and drivers for EVPN

L2VPN/VPLS CHALLENGES

- RFC7209: *Virtual Private LAN Service (VPLS), is a proven and widely deployed technology. However, the existing solution has a number of limitations when it comes to* ***redundancy, multicast optimization, and provisioning simplicity****. Furthermore, new applications are driving several new requirements for other L2VPN services such as Ethernet Tree (E-Tree) and Virtual Private Wire Service (VPWS).*

- Challenges:
  - All-Active redundancy mode for E-LAN and E-LINE (multihoming)
  - Multicast optimization
  - Provisioning simplicity
  - Network convergence upon failure independent of number of MAC address
  - Minimize the flooding of multi-destination frames (BUM)
  - Policy control over MAC address
  - E-TREE

DATA CENTER INDUSTRY CHALLENGES

DATACENTER INTERCONNECTION (DCI)

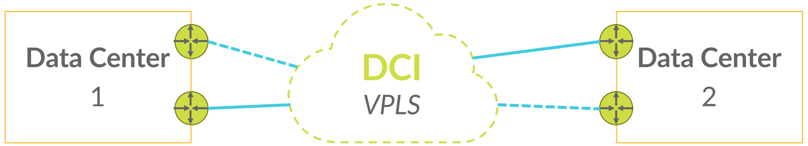
- No All-Active Forwarding
- No Control Plane Learning
- No Inter-Subnet Forwarding
- No MAC Mobility / Tromboning
- No Advanced Ethernet Services
  - VLAN-based
  - VLAN Bundle
  - VLAN Aware

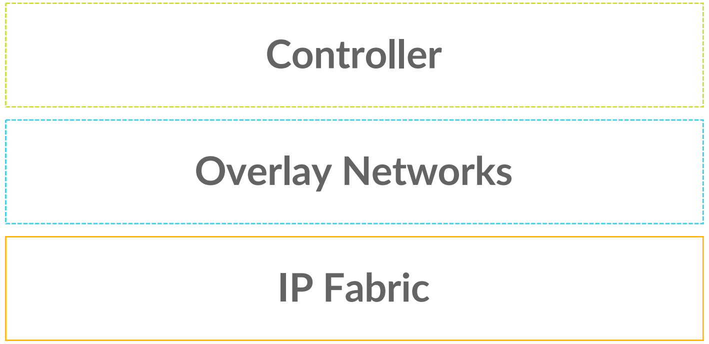
FABRIC INTERCONNECTION

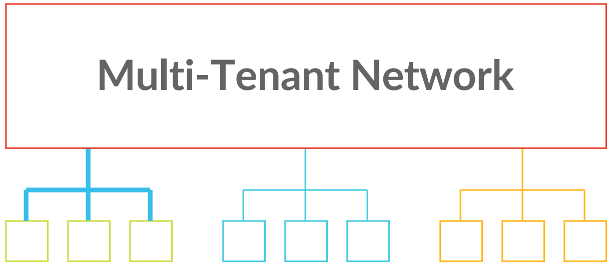
- No standardized control plane
- No standardized data plane
- Reinventing the wheel in many cases

MULTI-TENANCY NETWORK

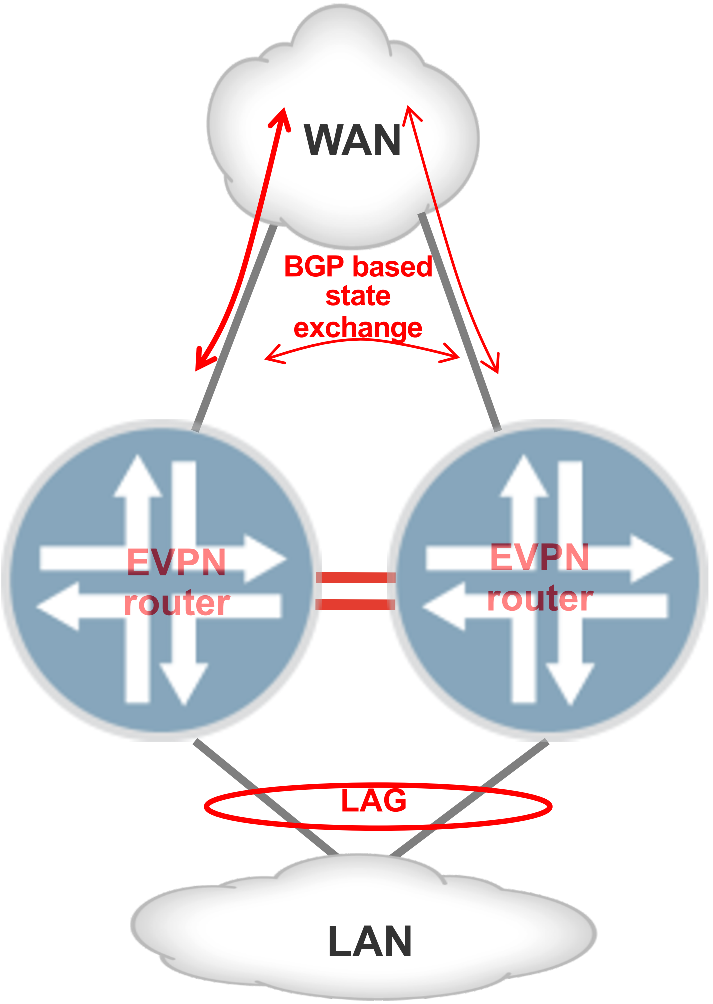
- No single solution
- Most implementations proprietary
- Complicated operations
- Limited scale
- Physical constraints – no L2 between PODs, etc







WHAT IS EVPN?



- ✓ **Next gen** L2VPN technology for **E-Line/E-LAN/E-Tree** Services
- ✓ Based on **open Standard** (Interoperable multi-vendor MC-LAG types of deployments)
- ✓ **EVPN control-plane for MPLS or IP** forwarding plane in modern Data Center architectures based in IP fabric
- ✓ **Datacenter Interconnectivity (DCI)** – for L2 or L3 service stretched (extended) between multiple DCs over WAN
- ✓ **Juniper leading** the multi-vendor industry wide initiative

EVPN APPLICATION

- Service providers
  - L2 E-Line/E-LAN/E-Tree services
    - EVPN technology improves their service offering
    - Operators can replace VPLS, H-VPLS, LDP PW with more efficient and unified technology
    - MPLS (LDP, RSVP, SPRING, BGP-LU) or IP (VxLAN, MPLSoUDP) underlay transport
  - L3 VPN Services
  - Multi-homed services
    - Operators can replace vendor proprietary (VC, MC-LAG) multi-homing solutions with standardized technology
    - Real multi-homing (not only dual-homing)
    - A/A type of deployments for L2 and/or L3 services
- Data Center Builders – SPs, Enterprises, Content providers
  - EVPN allows multi-tenant L2 service stretch between DCs
  - EVPN with VXLAN for L2 or L3 aware service stretch between VMs on an IP fabric DC

EVPN STANDARDS

- EVPN == BGP NLRI
  - is carried in BGP [RFC4271] using BGP Multiprotocol Extensions [RFC4760] with an Address Family Identifier (AFI) of 25 (L2VPN) and a Subsequent Address Family Identifier (SAFI) of 70 (EVPN). The NLRI field in the MP_REACH_NLRI/MP_UNREACH_NLRI attribute contains the EVPN NLRI.

- RFC7432 EVPN route types used in EVPN:
  - Auto Discovery route per Ethernet Segment (Type 1, AD per ESI)
  - Auto Discovery route per EVPN instance (Type 1, AD per EVI)
  - Mac address advertisement route (Type 2, MAC route)
  - Mac and IP addresses advertisement route (Type 2, MAC+IP route)
  - Inclusive Multicast Ethernet Tag route (Type 3, IMET route)
  - Ethernet segment route (Type 4)

EVPN STANDARDS

- Additional EVPN route types
  - IP Prefix advertisement
    - EVPN route type-5 - draft-ietf-bess-evpn-prefix-advertisement
  - Multicast optimizations:
    - Selective Multicast Ethernet Tag route (Type 6, SMET route)
    - IGMP Join Synch Route (Type 7)
    - IGMP Leave Synch Route (Type 8)
      - draft-ietf-bess-evpn-igmp-mld-proxy

- EVPN over IP transport (VxLAN, NVGRE, MPLS over GRE, GENEVE)
  - RFC 8365: A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)

EVPN VS VPLS

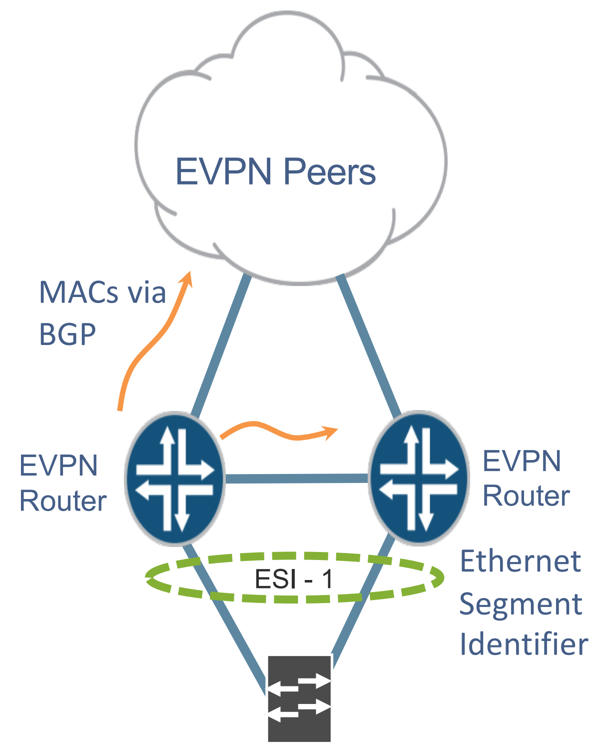
Function	VPN Functionality	VPLS	EVPN	Benefits
Address Learning	MAC address learning in the control plane (BGP)	✗	✔	Greater control and scalability through policies
Mobility	Hitless mobility	✗	✔	Near Hitless Host Mobility without renumbering L2 and L3 addresses
Redundancy	All Active Load Balancing Fast Convergence	✗	✔	Active redundancy Optimized link utilization
Traffic Optimization	Default Gateway Synchronization ARP/ND Proxy MAC Mobility	✗	✔	Optimized L2 & L3 traffic flows Reduced BUM flooding
Provisioning	L2 and L3 over same interface	✗	✔	Simplified provisioning
Data Plane Options	Different types of data plane	✗	✔	Allows non-MPLS data plane (IP data plane)

EVPN

Started off as a DC technology but is gaining momentum in all 5 domains

Key EVPN advantages over all the domains:
- Active / Active multi-homing
- Load balancing
- Layer 3 integration
- Faster error recovery
- Finer grain policy control due to BGP
- VM mobility (especially for DC)

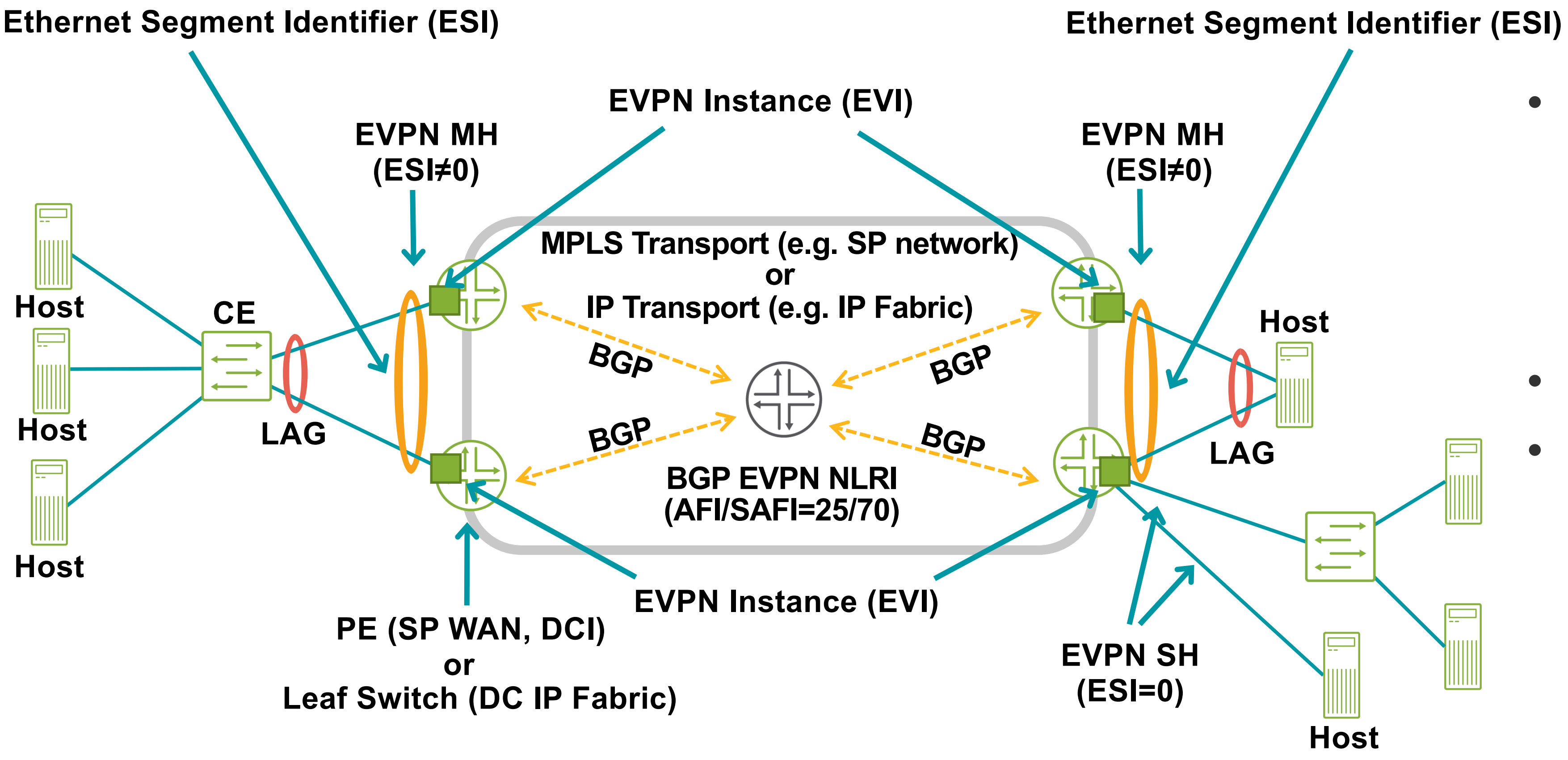
Data Center Gateway	All Active load balancing. Seamless L2/L3 Interconnect
Business Edge	Multi-homed, feature rich E-LINE, E-LAN E-TREE
Metro	Multi-homing and L3 Integration
Peering	Policy driven peering relationships
OTT	L2/L3 Services over simple IP connectivity



EVPN

Architectural model and building blocks

EVPN ARCHITECTURAL MODEL – RFC7432



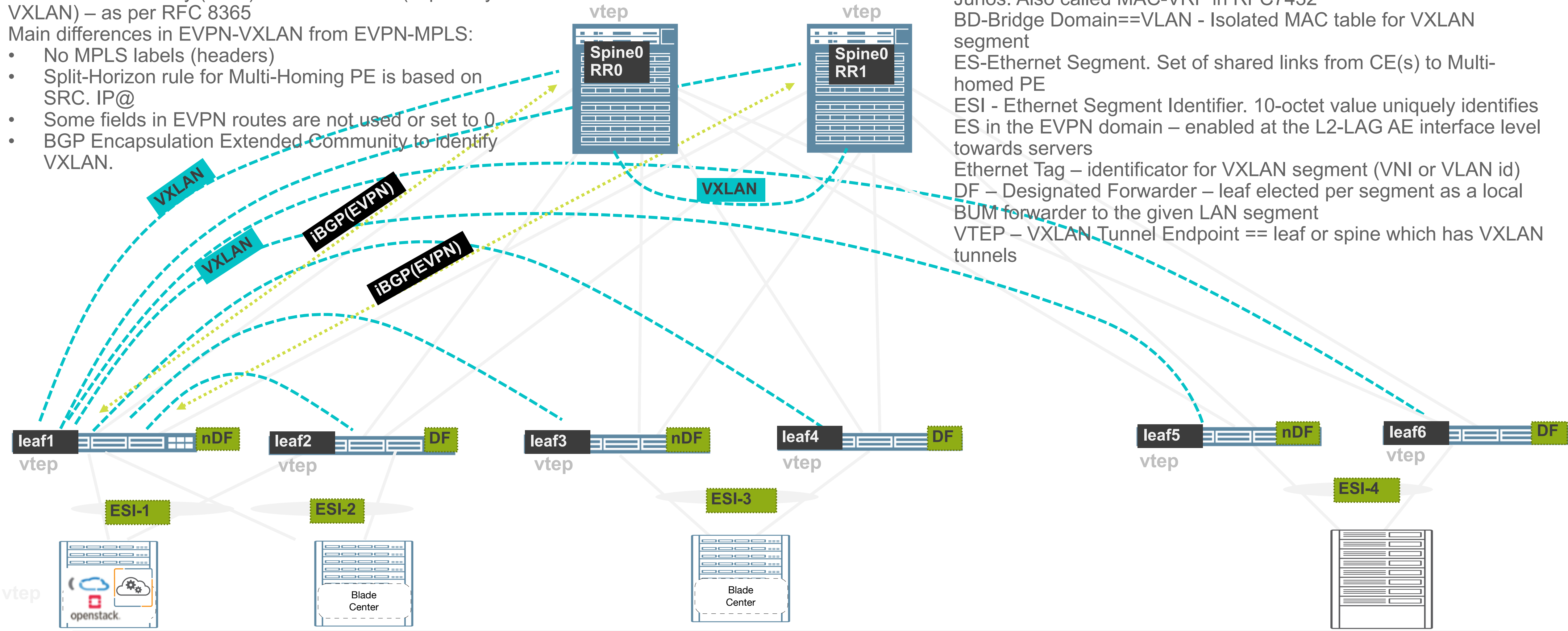
- BGP Control Plane
  - Policy Control (like L3VPN)
  - Minimizes BUM flooding
- IP or MPLS transport
- EVPN Multi-homing
  - All-active (A/A)
  - Single-active (A/S)

EVPN NETWORK VIRTUALIZATION OVERLAY (NVO) – RFC8365

EVPN RFC7432 standard was adapted to use as Network Virtualization Overlay (NVO) solution over IP (especially VXLAN) – as per RFC 8365
Main differences in EVPN-VXLAN from EVPN-MPLS:

- No MPLS labels (headers)
- Split-Horizon rule for Multi-Homing PE is based on SRC. IP@
- Some fields in EVPN routes are not used or set to 0
- BGP Encapsulation Extended Community to identify VXLAN.

EVI - EVPN Instance - Routing-Instance type virtual-switch in Junos. Also called MAC-VRF in RFC7432
BD-Bridge Domain==VLAN - Isolated MAC table for VXLAN segment
ES-Ethernet Segment. Set of shared links from CE(s) to Multi-homed PE
ESI - Ethernet Segment Identifier. 10-octet value uniquely identifies ES in the EVPN domain – enabled at the L2-LAG AE interface level towards servers
Ethernet Tag – identificator for VXLAN segment (VNI or VLAN id)
DF – Designated Forwarder – leaf elected per segment as a local BUM forwarder to the given LAN segment
VTEP – VXLAN Tunnel Endpoint == leaf or spine which has VXLAN tunnels



DATA PLANE META DATA
EVPN OVER MPLS VS EVPN OVER VXLAN

MPLS	Transport Label	Service Label	Payload
VXLAN	Outer IP Header	VXLAN VNID	Payload

- Forwarding
- Service Separation

- Split Horizon
- Hashing

WHY VXLAN

Where is VxLAN Applicable:

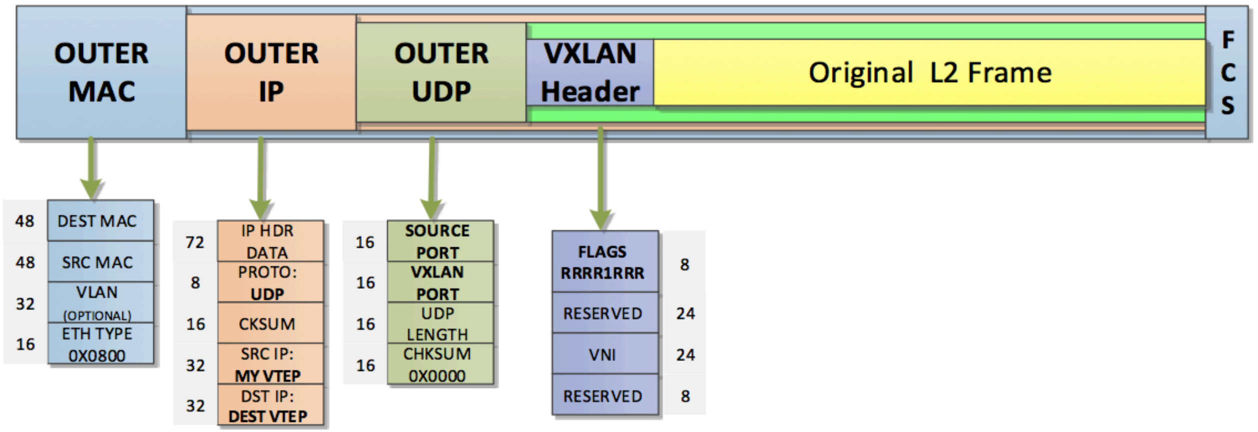
- L2 connectivity for VMs in a fully flat IP fabric based datacenters
  - Run IGP- OSPF, ISIS
  - No MPLS requirements
  - Eliminates spanning tree, l2 loop/guard issues
  - Cheap L2 devices capable of supporting small DC FIB requirement
- SDN enabled Datacenters

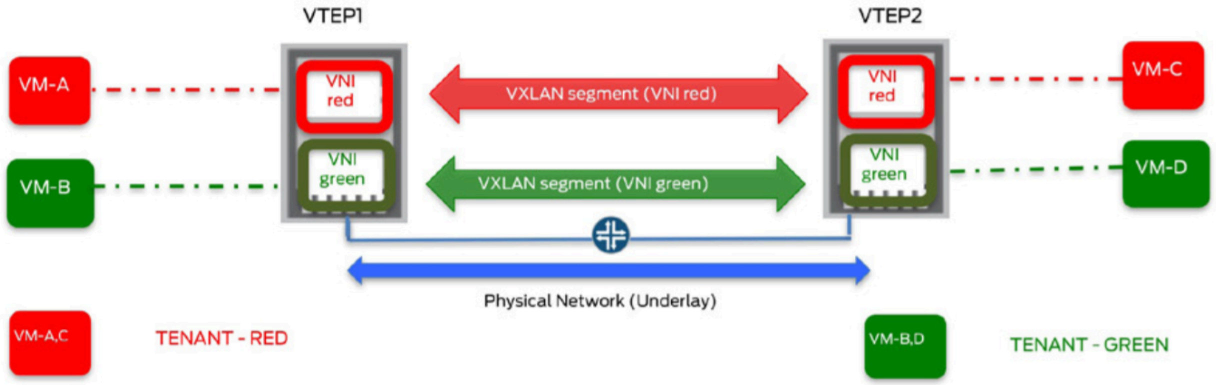
Who will be interested in VXLAN :

- Cloud Builders – SPs, Enterprises, Content providers
  - The DC builders that design fully flat IP fabric based DCs
- Enterprises that want to run Over The Top (OTT) L2 VPN connectivity on an IP transport network
  - VXLAN as a transport
  - Similar to MPLS over GRE model, but VXLN is entropy friendly compared to GRE

EVPN/VXLAN – VXLAN TUNNEL TRANSPORT

- Once the EVPN signaling is completed the VXLAN tunnels are automatically created using the global routing table lo0.0 IP@ as source
- Virtual eXtensible Local Area Network (**VXLAN**) defines in RFC 7348
- **VXLAN encapsulation** is also called MAC-over-UDP
  - Original Ethernet frame(without FCS) is encapsulated into VXLAN Header + UDP Header
- VXLAN standard defines following terms:
  - **hw_vtep** - VXLAN Tunnel EndPoint. Termination point of VXLAN tunnels on leaf/spine, typically associated with lo0 interface
  - **VXLAN Segment** – Broadcast Ethernet segment connected via VXLAN tunnels (same as VLAN segment or L2 segment)
  - **VNI** - VXLAN Network Identifier (VLAN ID)
  - **UDP Destination port** is always **4789**
- **UDP Source port** is in range **49152-65535**. Actual value is calculated using **hashing algorithm** from L2/L3/L4 headers of original headers
- **MTU** 1554 for an original frame 1500 frame:
  - IPv4 header: 20 bytes
  - UDP header: 8 bytes
  - VXLAN header: 8 bytes
  - Original Ethernet with tagging: 14 bytes + 4 if vlan tagged

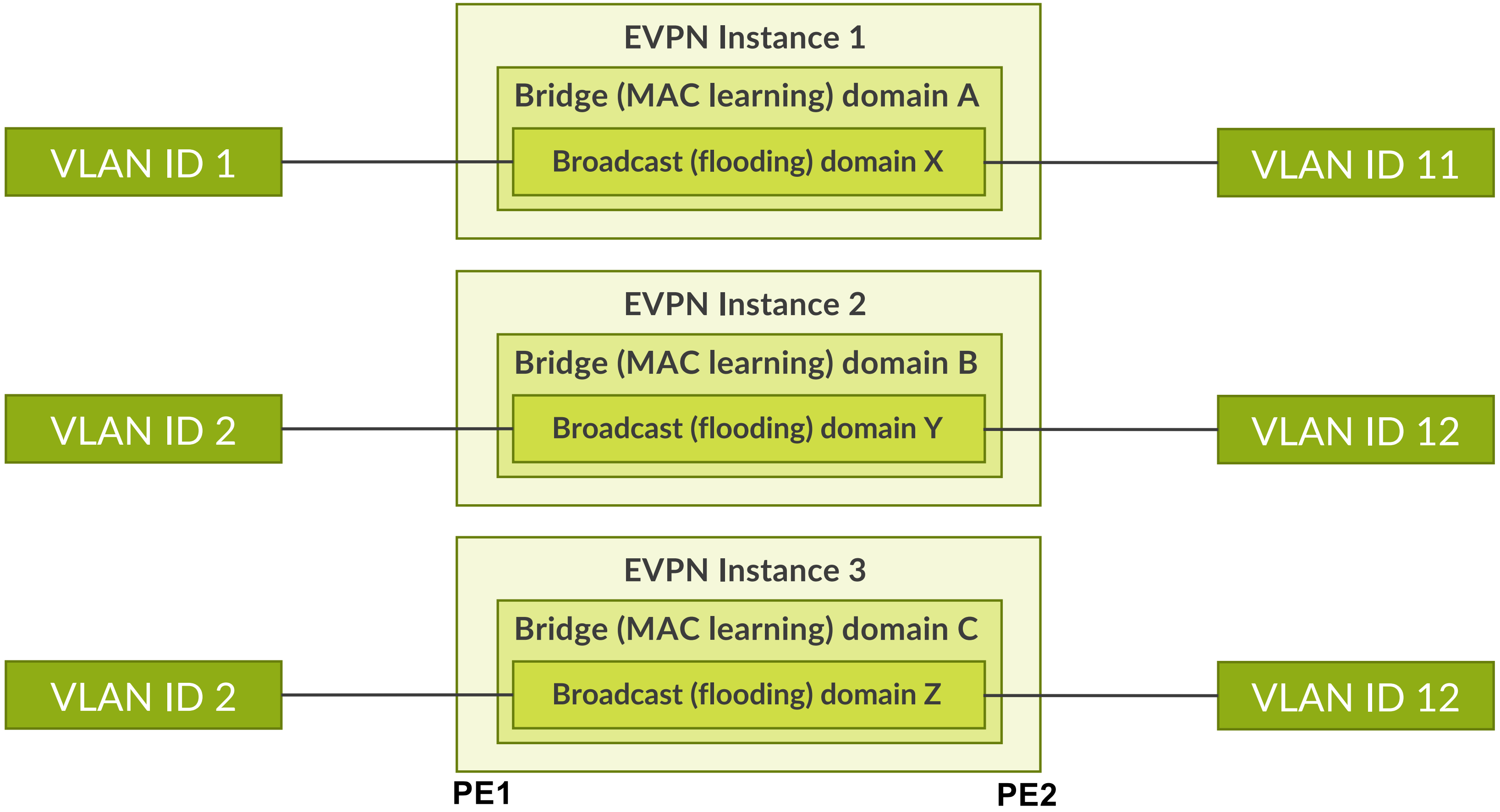




EVPN

Service Types

EVPN E-LAN VLAN-BASED



Single BD per EVI

Single VLAN per BD

VLAN translation possible

VLAN ID can be stripped or carried across the EVPN backbone

Ethernet-tag ID = 0

EVPN E-LAN VLAN-BASED – RFC 7432, SECTION 6.1

Scenario 1: The same CE VLAN ID used on all PE (thus, VLAN translation not required)

- RFC 7432, Section 6.1 doesn't specify, if VLAN ID should be carried or not
  - Configuration Option 1 (VLAN ID carried)
  - Configuration Option 2 (VLAN ID not carried)

Scenario 2: Different CE VLAN ID used on PEs (thus, VLAN translation required)

- RFC 7432, Section 6.1 specifies, that original VLAN ID *SHOULD* be carried across EVPN backbone, and VLAN translation *MUST* occur at egress PE
  - Configuration Option 3: conforms to SHOULD and MUST
  - Configuration Option 4: conforms to MUST only

In all cases, The Ethernet Tag ID in all EVPN routes MUST be set to 0

- Fulfilled by configuration options 1 trough 4

EVPN E-LAN VLAN-BASED

Option 1: No VLAN translation, VLAN ID not carried across EVPN backbone

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;          <========= the same CE-VID used at all PEs
        }
    }
}
routing-instances {
    VLAN-BASED-NO-VID {
        instance-type evpn;
        vlan-id none; <================== originating VID removed, Ethernet Tag ID = 0
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

EVPN E-LAN VLAN-BASED

Option 2: No VLAN translation, VLAN ID carried across EVPN backbone

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;          <========= the same CE-VID used at all PEs
        }
    }
}
routing-instances {
    VLAN-BASED-WITH-VID {
        instance-type evpn;
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

EVPN E-LAN VLAN-BASED

Option 3: VLAN translation, VLAN ID carried across EVPN backbone

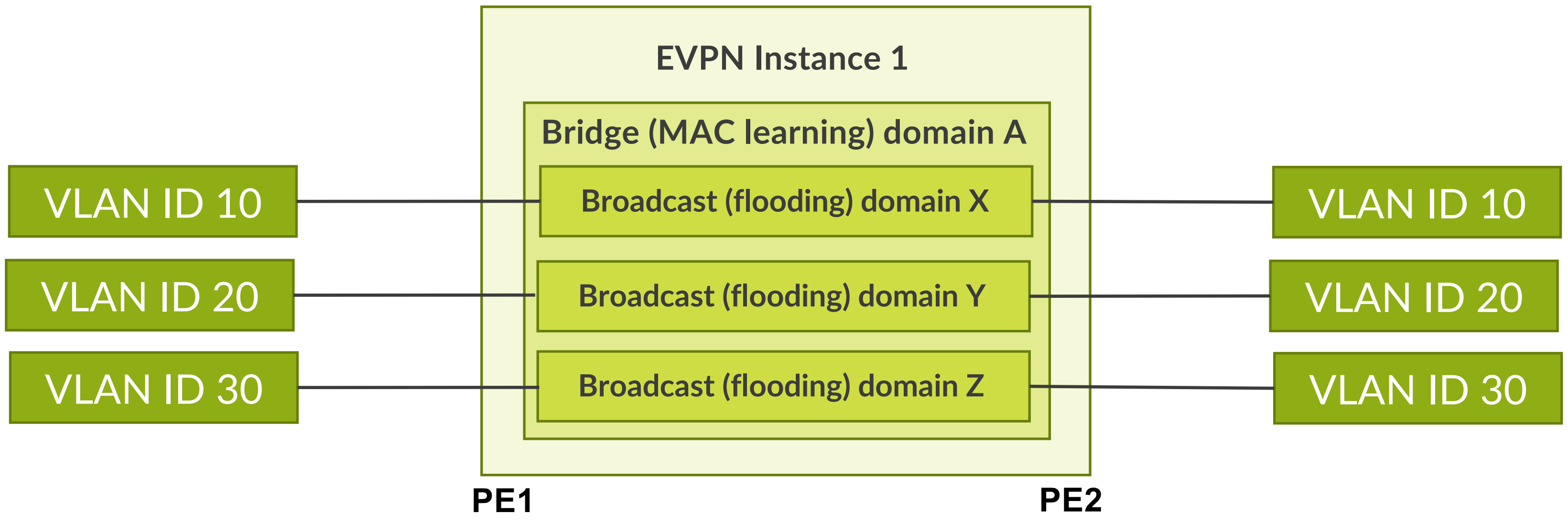
```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;          <========== different CE-VID might be used at different PEs
            output-vlan-map {     <========== translation at disposition (egress) PE
                swap;
            }
        }
    }
}
routing-instances {
    VLAN-BASED-WITH-NORMALIZATION-STRICT-RFC-COMPLIANCE {
        instance-type evpn;
        vlan-id none;     <=================== Ethernet Tag ID = 0
        no-normalization; <=================== original VID remained
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

EVPN E-LAN VLAN-BASED

Option 4: VLAN translation, no VLAN ID carried across EVPN backbone

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;          <========= different CE-VID might be used at different PEs
        }                         <========= no explicit out vlan-map, translation (from no VID
    }                             <========= to VID on IFL) will happen automatically
}
routing-instances {
    VLAN-BASED-WITH-NORMALIZATION-LOOSE-RFC-COMPLIANCE {
        instance-type evpn;
        vlan-id none;          <================== originating VID removed, Ethernet Tag ID = 0
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

EVPN E-LAN VLAN-BUNDLE



Single BD per EVI

Multiple VLANs per BD

VLAN translation not possible

VLAN ID is carried across the EVPN backbone

Ethernet-tag ID = 0

EVPN E-LAN VLAN-BUNDLE

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;
        }
        unit 101 {
            encapsulation vlan-bridge;
            vlan-id 101;
        }
    }
}
routing-instances {
    VLAN-BUNDLE {
        instance-type evpn;
        interface xe-0/3/0.100;
        interface xe-0/3/0.101;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

Different CE VLAN IDs on IFLs

No VLAN normalization

- No vlan-id at instance level

VLANs carried across the EVPN services

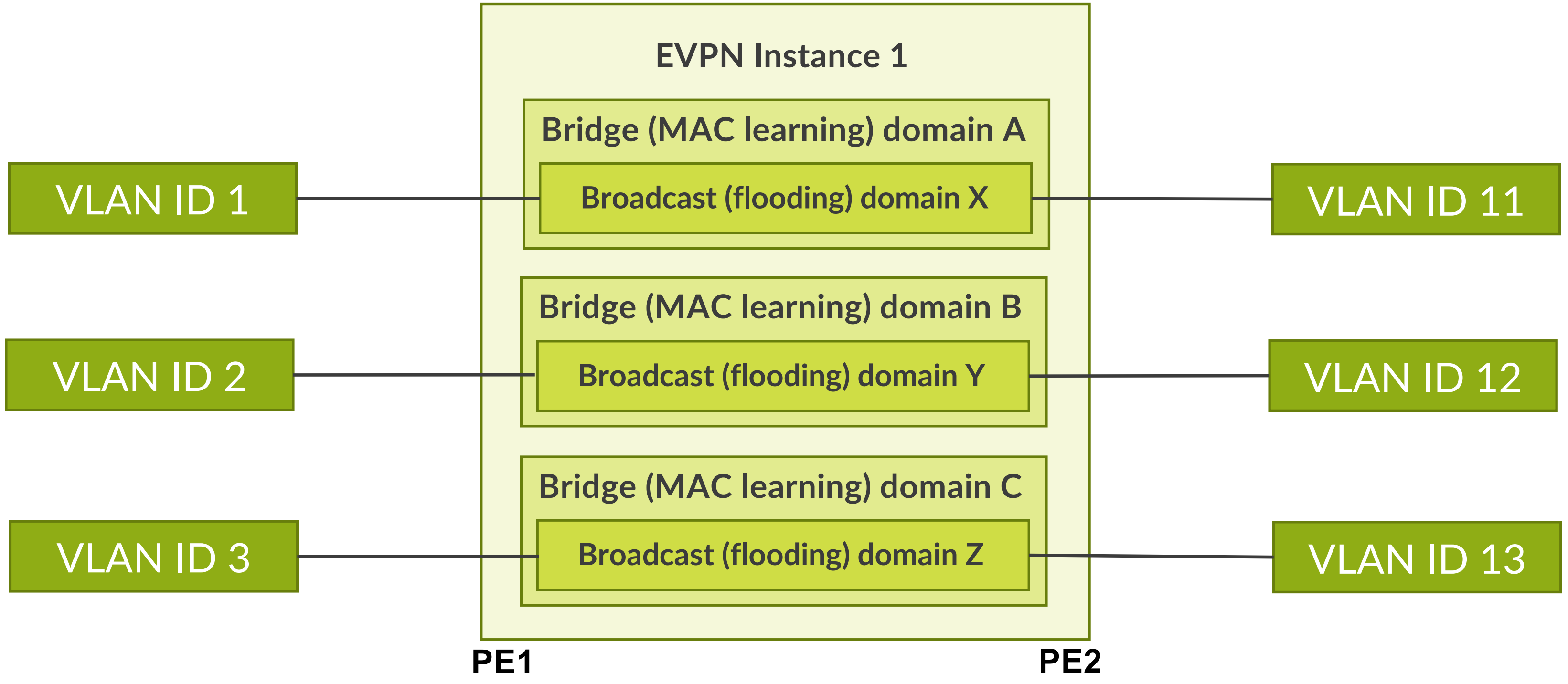
EVPN E-LAN VLAN-BUNDLE

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id-list [ 100-101 ];          <========= CE-VID 100, 101
        }
    }
}
routing-instances {
    VLAN-BUNDLE {
        instance-type evpn;
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

Better scaling

- Smaller number of IFLs required
- Smaller configuration
- Faster commits in large scale deployments

EVPN E-LAN VLAN-AWARE BUNDLE



Multiple BDs per EVI

Single VLAN per BD

VLAN translation possible (VLAN normalization)

VLAN ID is carried across the EVPN backbone

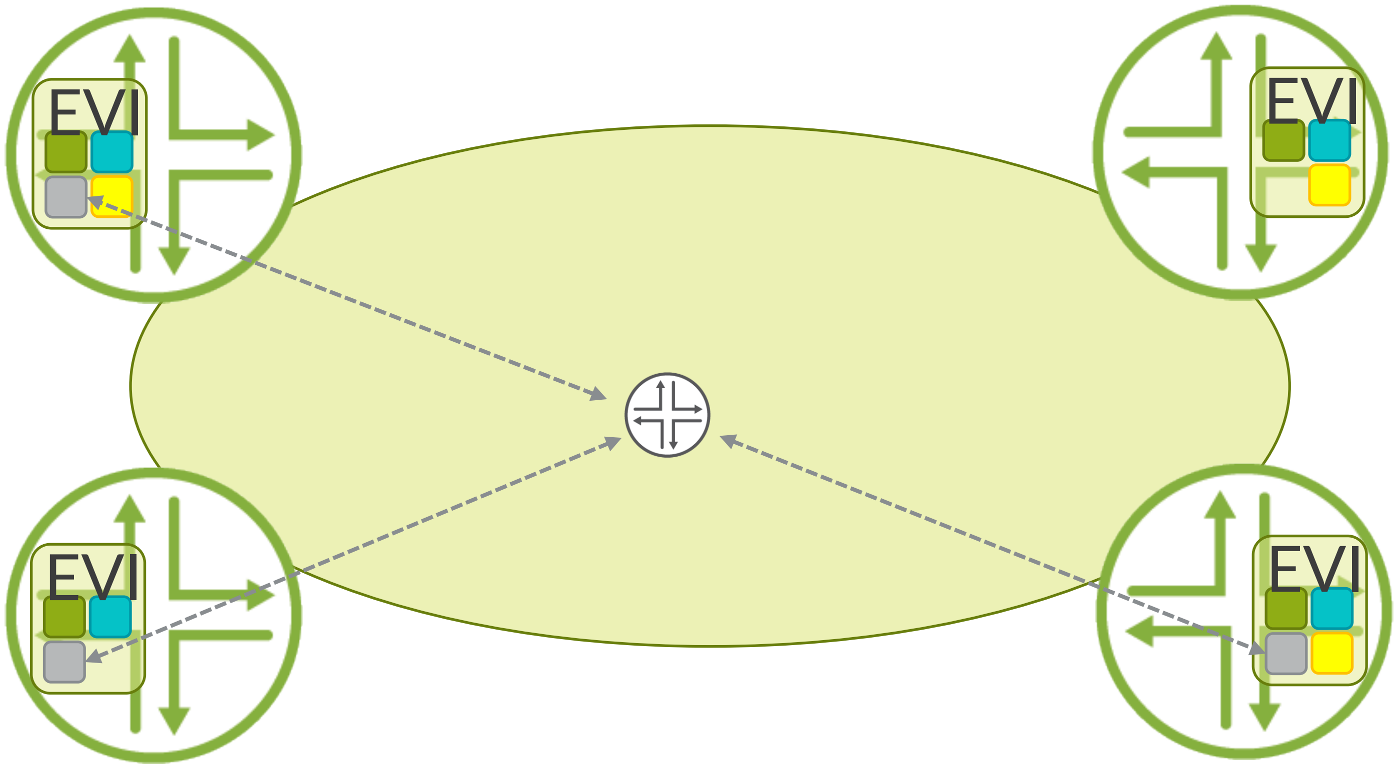
Ethernet-tag ID = normalized VLAN

EVPN E-LAN VLAN-AWARE-BUNDLE

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;
        }
        unit 101 {
            encapsulation vlan-bridge;
            vlan-id 101;
        }
    }
}
```

```
routing-instances {
    VLAN-AWARE-BUNDLE {
        instance-type virtual-switch;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn extended-vlan-list 100-101;
        bridge-domains {
            BD-100 {
                domain-type bridge;
                vlan-id 100;
                interface xe-0/3/0.100;
            }
            BD-101 {
                domain-type bridge;
                vlan-id 101;
                interface xe-0/3/0.101;
            }
        }
    }
}
```

EVPN E-LAN VLAN-AWARE-BUNDLE (OPTIMIZATION)



There are scenarios, where not all BDs belonging to the same VLAN-aware bundle service are present on all PEs

- Inefficient to send BD scoped EVPN Routes to all PEs:
  - MAC/IP – Type 2
  - IMET – Type 3
  - SMET – Type 6
- Optimization
  - automatically allocate different RT for each BD
  - when using together with RT constraints, limits distribution of MAC/IP (Type 2) and IMET (Type 3) routes to required PEs only

EVPN E-LAN VLAN-AWARE-BUNDLE (OPTIMIZATION)

RT for each BD can be automatically generated in following format

- target:<local-AS>:<Eth-Tag-ID>

- Non BD-scoped EVPN Routes (e.g. E-AD/EVI – Type 1) have their RT manually assigned by VRF export policy

Domain ID (0-15) can be used to distinguish overlapped VLANs

- RFC 8365, Section 5.1.2.1

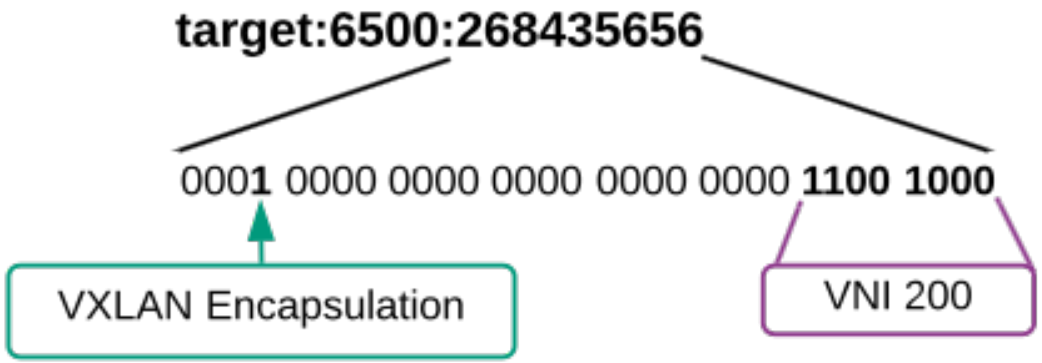
```
routing-instances {
    VLAN-AWARE-BUNDLE {
        instance-type virtual-switch;
        route-distinguisher 10.0.0.1:100;
        vrf-target {
            target:65303:101100;     # E-AD/EVI (Type 1)
            auto;       # MAC/IP (Type 2) + IMET (Type3)
        }
        protocols evpn extended-vlan-list 100-101;
        bridge-domains {
            BD-100 {
                domain-type bridge;
                vlan-id 100;
                interface xe-0/3/0.100;
                domain-id 1;                    # Optional
            }
            BD-101 {
                domain-type bridge;
                vlan-id 101;
                interface xe-0/3/0.101;
                domain-id 1;                    # Optional
            }
        }
    }
}
```

EVPN E-LAN VLAN-AWARE-BUNDLE (OPTIMIZATION)

```
* 2:192.168.0.2:201::201::56:68:a3:1e:2c:2d/304 MAC/IP (1 entry, 1 announced)
 BGP group IBGP-TO-RR type Internal
     Route Distinguisher: 192.168.0.2:201
     Route Label: 96
     ESI: 00:11:22:33:44:55:66:00:00:00
     Nexthop: Self
     Localpref: 100
     AS path: [65000] I
     Communities: target:65000:251658441
```

0xF0000C9

Domain ID: 15 VLAN ID: 201



EVPN E-LAN VLAN-AWARE-BUNDLE (SPECIAL CASE)

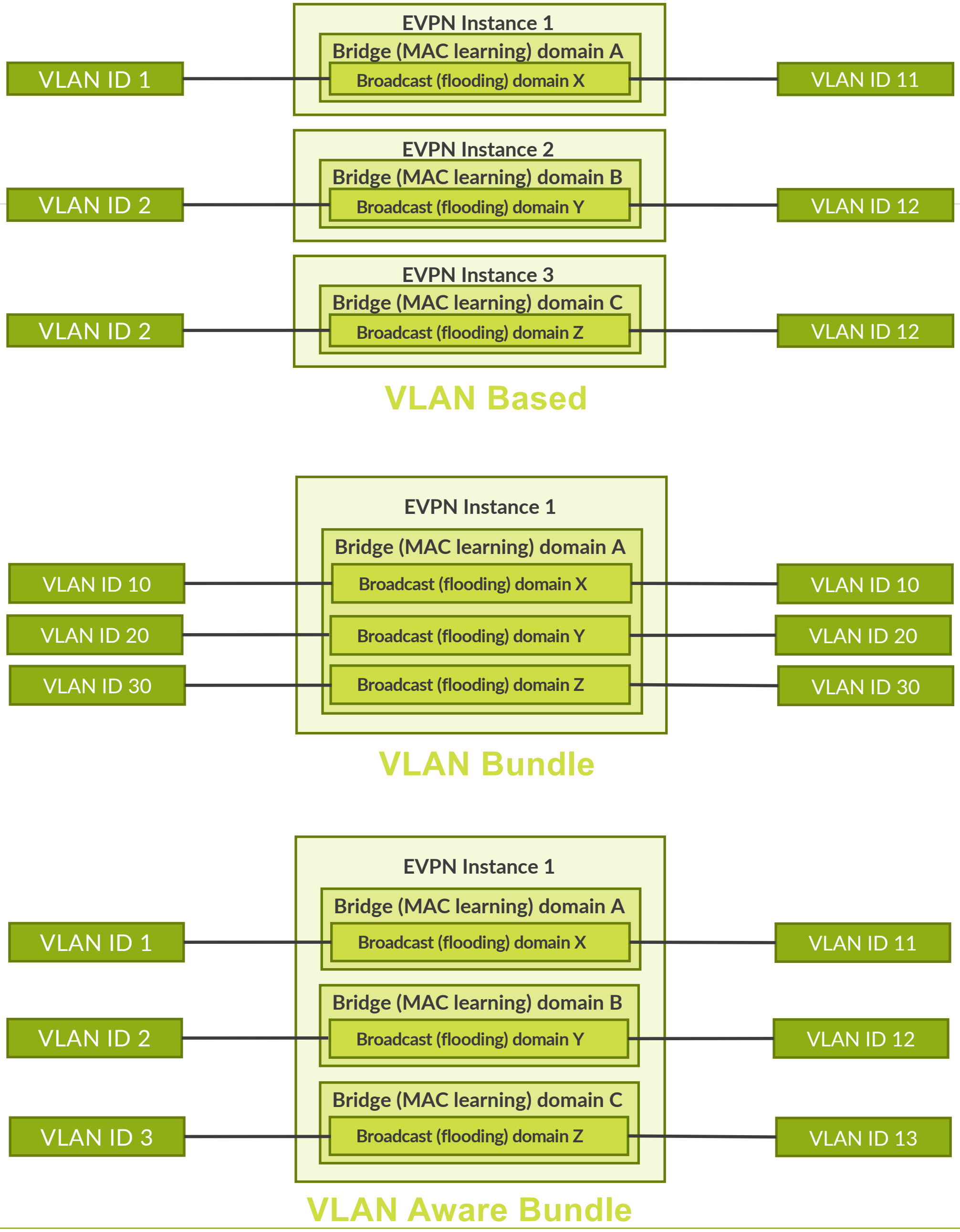
Special case of VLAN-aware bundle (i.e. Ethernet Tag is not ‘0’) with single BD in the instance

```
interfaces {
    xe-0/3/0 {
        unit 100 {
            encapsulation vlan-bridge;
            vlan-id 100;          <========= different CE-VID might be used at different PEs
        }                         <========= no explicit out vlan-map, translation happens
    }                             <========= automatically via VLAN normalization
}
routing-instances {
    VLAN-AWARE-BUNDLE-SINGLE-BD {
        instance-type evpn;
        vlan-id 200;          <================== normalized VLAN, Ethernet Tag ID = 200
        interface xe-0/3/0.100;
        route-distinguisher 10.0.0.1:100;
        vrf-target target:65303:101100;
        protocols evpn;
    }
}
```

EVPN SERVICE TYPES

Summary

Characteristics	VLAN Based	VLAN Bundle	VLAN Aware Bundle
Junos routing instance type	evpn	evpn	virtual-switch
Broadcast domains (VLANs) per EVI	1	>1	>1
Bridge (MAC learning) domains per EVI	1	1	>1
MACs must be unique across VLANs	no	yes	no
VLAN translation allowed?	yes	no	yes
VLAN tag carried over?	no/yes	yes	yes
Ethernet Tag (BD) ID on BD scoped routes	0	0	≠0
Port based service support	no	yes	yes



EVPN

Route types

EVPN NLRI ROUTE TYPES

EVPN route types are discussed later in details

Type	Scope	Route Name	Standardization	Functional description
Type 1	ESI	Ethernet Auto-discovery Route	RFC 7432	Used for advertising split-horizon label (common label for all EVIs on ESI) and to enable fast convergence (mass withdrawal).
Type 1	EVI BD	Ethernet Auto-discovery Route	RFC 7432	Used for advertising the EVPN aliasing label (distinct label for each EVI)
Type 2	BD	MAC/IP Advertisement Route	RFC 7432	Used for announcing service label for reachability for a MAC address
Type 3	BD	Inclusive Multicast Ethernet Tag Route	RFC 7432	Sets up paths for BUM traffic per VLAN per EVI basis This route might be filtered to block BUM traffic
Type 4	ESI	Ethernet Segment Route	RFC 7432	Allows PEs with same ESI discover each other Used for Designated Forwarder (DF) Election
Type 5	VRF	IP Prefix Route	draft-ietf-bess-evpn-prefix-advertisement	Used for advertising L3 prefix information (L3VPN)
Type 6	BD	Selective Multicast Ethernet Tag Route	draft-ietf-bess-evpn-igmp-mld-proxy	Used for advertising IGMP/MLD Membership messages (IGMP Proxy) Reduces IGMP/MLD Group Membership flooding Prevents sending MCAST traffic to EVPN PEs with no MCAST receivers
Type 7	ESI	Multicast Join Sync Route	draft-ietf-bess-evpn-igmp-mld-proxy	Used for synchronizing IGMP/MLD Group Membership states between PEs connected to common ESI
Type 8	ESI	Multicast Leave Sync Route	draft-ietf-bess-evpn-igmp-mld-proxy	Used for synchronizing IGMP/MLD Group Membership states between PEs connected to common ESI

EVPN NEW CONCEPT: ETHERNET TAG IDENTIFIER

An Ethernet Tag ID is a 32-bit field, used in EVPN BGP NLRIs (Control Plane)

Contains a 12-bit or a 24-bit identifier to identify a bridge-domain in an EVPN VLAN-aware bundle instance (e.g. in MAC/IP – Type 2, IMET – Type 3, or SMET – Type 6 EVPN routes)

- 12-bit identifier is used for normalized bridge-domain VLAN ID for EVPN-MPLS
- 24-bit identifier is used for VNID for EVPN-VxLAN

Set to ‘0’ for non-BD scoped EVPN Route types:

- All EVPN routes for VLAN-based or VLAN-bundle services

An EVI can have one or more bridge-domains assigned to a given EVPN instance

EVPN NLRI ROUTE TYPES (ESI SPECIFIC)

Type 1 → Ethernet Auto-Discovery (AD) Route
RD → <router-id>:0
ESI → leading ‘0’-s abbreviated with ‘::’
Eth-Tag ID → all ‘FF’ in E-AD/ESI route

```
1:192.168.0.2:0::0201220199000000::FFFF:FFFF/192 AD/ESI
```

Type 4 → Ethernet Segment (ES) Route
RD → <router-id>:0
ESI → leading ‘0’-s abbreviated with ‘::’
Originator’s IP → <router-id>

```
4:192.168.0.2:0::0201220199000000:192.168.0.2/296 ES
```

Route Type 1 (AD) per ESI and Type 4 (ES) are generated only for multi-homed Ethernet Segments, i.e. when ESI≠0.

Configured (or derived from loopback) Router-ID is used in RD and Originator’s IP fields

Route Type 1 (AD) has all RTs of all EVIs attached to given ESI

EVPN NLRI ROUTE TYPES (EVI SPECIFIC)

- Type 1 → Ethernet Auto-Discovery (AD) Route
- RD → configured in EVI (or auto-derived)
- ESI → leading ‘0’-s abbreviated with ‘::’
- Eth-Tag ID → ’0’ (E-LAN/E-Tree), non-’0’ (E-Line)

```
1:192.168.0.2:201::0201222222000000::0/192 AD/EVI
```

- Type 2 → MAC/IP Advertisement (MAC/IP) Route
- RD → configured in EVI (or auto-derived)
- Eth-Tag ID → ‘0’ (based/bundle) or non-’0’ (aware-bundle)
- MAC → learned from data plane

```
2:192.168.0.2:201::201::2c:6b:f5:41:05:c0/304 MAC/IP
```

- Type 2 → MAC/IP Advertisement (MAC/IP) Route
- RD → configured in EVI (or auto-derived)
- Eth-Tag ID → ‘0’ (based/bundle) or non-’0’ (aware-bundle)
- MAC → learned from data plane
- IP Address → learned by ARP proxy feature

```
2:192.168.0.2:201::201::2c:6b:f5:41:05:c0::10.2.1.2/304 MAC/IP
```

Note: ESI not shown in basic show command for Type 2 (MAC/IP) Routes

EVPN NLRI ROUTE TYPES (EVI SPECIFIC)

Type 3 → Inclusive Multicast Ethernet Tag (IM) Route
RD → configured in EVI (or auto-derived)
Eth-Tag ID → '0' (based/bundle) or non-'0' (aware-bundle)
Originator's IP → <router-id>

```
3:192.168.0.2:201::201::192.168.0.2/248 IM
```

Type 6 → Selective Multicast Ethernet Tag (SMET) Route
RD → configured in EVI (or auto-derived)
Eth-Tag ID → '0' (based/bundle) or non-'0' (aware-bundle)
MCAST Group ID
Originator's IP → <router-id>

```
6:192.168.0.1:201::201::233.1.1.1::3.3.3.3/520
```

EVPN NLRI ROUTE TYPES (ESI SPECIFIC)

- Type 7 → Multicast Join Sync Route (*,G)
- RD → configured in EVI (or auto-derived)
- ESI → leading ‘0’-s abbreviated with ‘::’
- Eth-Tag ID → ‘0’ (based/bundle) or non-’0’ (aware-bundle)
- MCAST Group ID
- Originator’s IP → <router-id>

```
7:192.168.0.1:201::0201222222000000::201::233.1.1.1::3.3.3.3/600
```

- Type 7 → Multicast Join Sync Route (S,G)
- RD → configured in EVI (or auto-derived)
- ESI → leading ‘0’-s abbreviated with ‘::’
- Eth-Tag ID → ‘0’ (based/bundle) or non-’0’ (aware-bundle)
- MCAST Source
- MCAST Group ID
- Originator’s IP → <router-id>

```
7:192.168.0.1:201::0201222222000000::201::1.2.3.4::233.1.1.1::3.3.3.3/600
```

EVPN NLRI ROUTE TYPES (ESI SPECIFIC)

- Type 8 → Multicast Leave Sync Route (*,G)
- RD → configured in EVI (or auto-derived)
- ESI → leading '0'-s abbreviated with '::'
- Eth-Tag ID → '0' (based/bundle) or non-'0' (aware-bundle)
- MCAST Group ID
- Originator's IP → <router-id>

```
8:192.168.0.1:201::0201222222000000::201::233.1.1.1::3.3.3.3/600
```

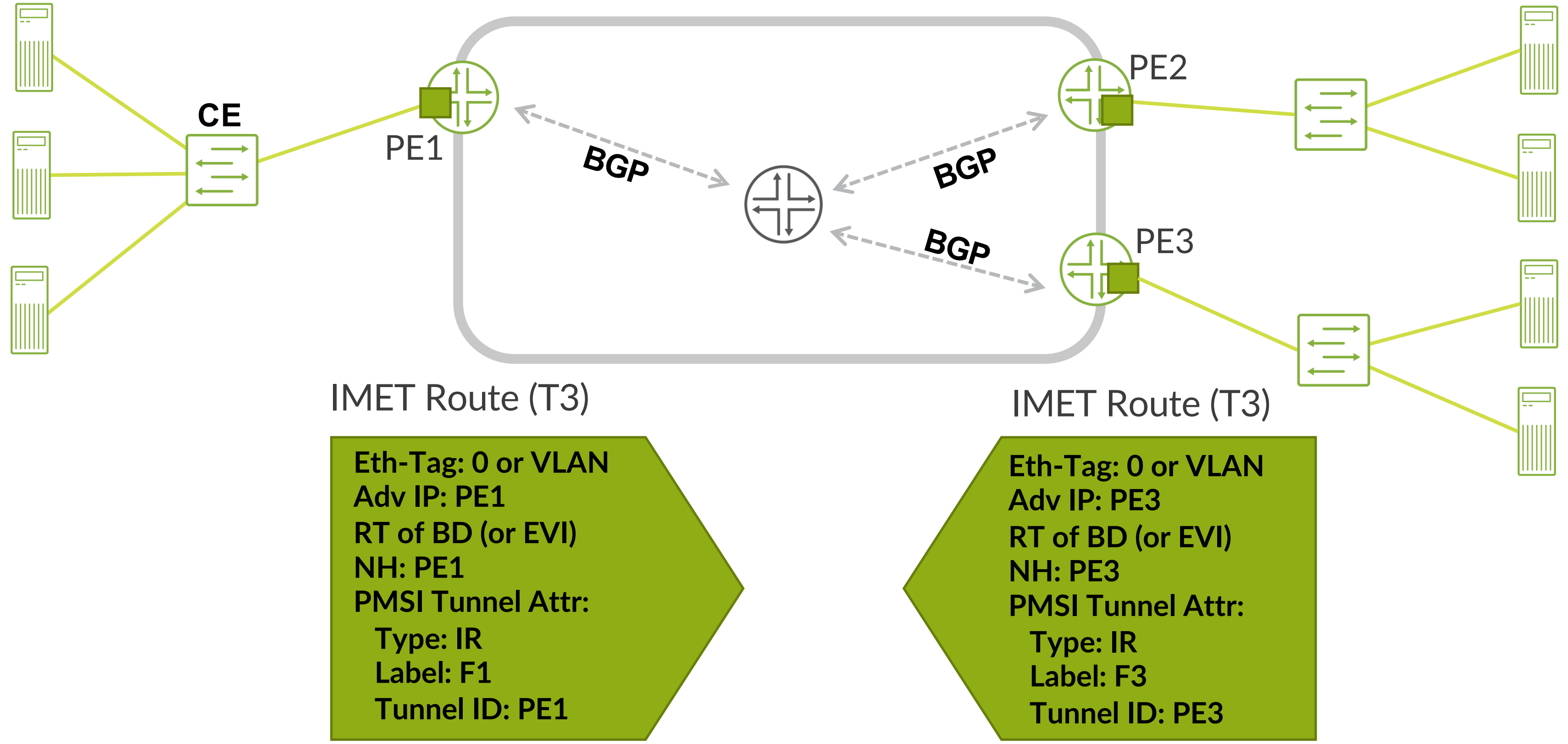
- Type 8 → Multicast Leave Sync Route (S,G)
- RD → configured in EVI (or auto-derived)
- ESI → leading '0'-s abbreviated with '::'
- Eth-Tag ID → '0' (based/bundle) or non-'0' (aware-bundle)
- MCAST Source
- MCAST Group ID
- Originator's IP → <router-id>

```
8:192.168.0.1:201::0201222222000000::201::1.2.3.4::233.1.1.1::3.3.3.3/600
```

EVPN

Operations

EVPN OPERATION: SETUP BUM PATH (INGRESS REPLICATION)



Each PE advertises IMET (T3) route

- Per BD (VLAN-aware bundle)
- Per EVI (VLAN-based, VLAN-bundle)

IMET (T3) route contains PMSI (P-Multicast Service Interface) with info required to flood BUM

EVPN OPERATION: SETUP BUM PATH (INGRESS REPLICATION)

Ethernet Tag (BD) ID

Originator IP: <router-id>

```
root@R1# run show route advertising-protocol bgp 192.168.0.4 match-prefix 3:* table RI-EVPN-22301.evpn.0 detail

RI-EVPN-22301.evpn.0: 10 destinations, 10 routes (10 active, 0 holddown, 0 hidden)
* 3:192.168.0.1:22301::301::192.168.0.1/248 IM (1 entry, 1 announced)
 BGP group IBGP-TO-RR type Internal
     Route Distinguisher: 192.168.0.1:22301
     Route Label: 100048
     PMSI: Flags 0x0: Label 100048: Type INGRESS-REPLICATION 192.168.0.1
     Nexthop: Self
     Localpref: 100
     AS path: [65000] I
     Communities: 65199:65199 target:65000:22301
     PMSI: Flags 0x0: Label 100048: Type INGRESS-REPLICATION 192.168.0.1
```

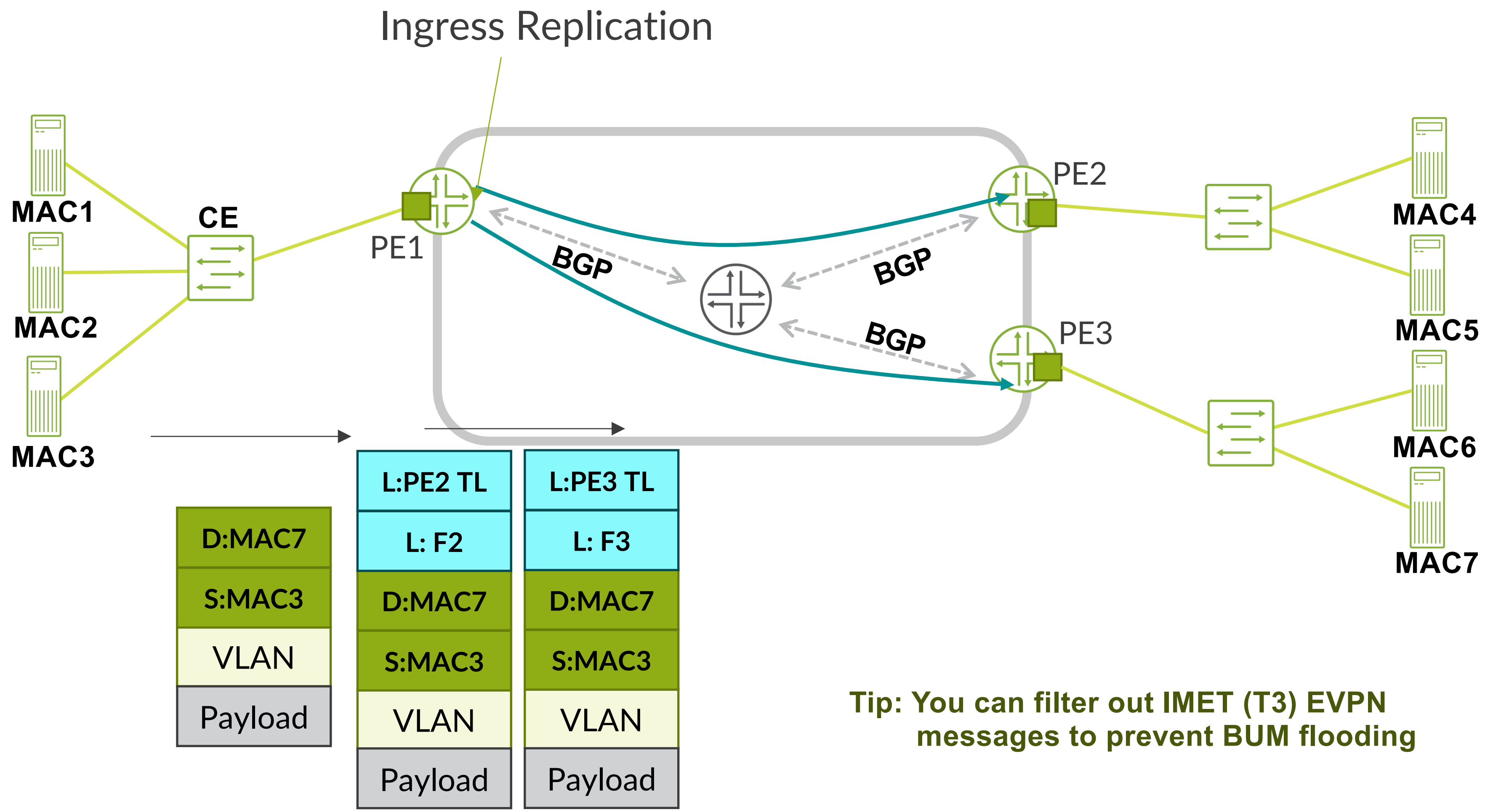
Next-Hop IP: <router-id>

Flood label

RT (per BD or per EVI)

Tunnel End Point: <router-id>

EVPN OPERATION: BUM FLOODING (INGRESS REPLICATION)

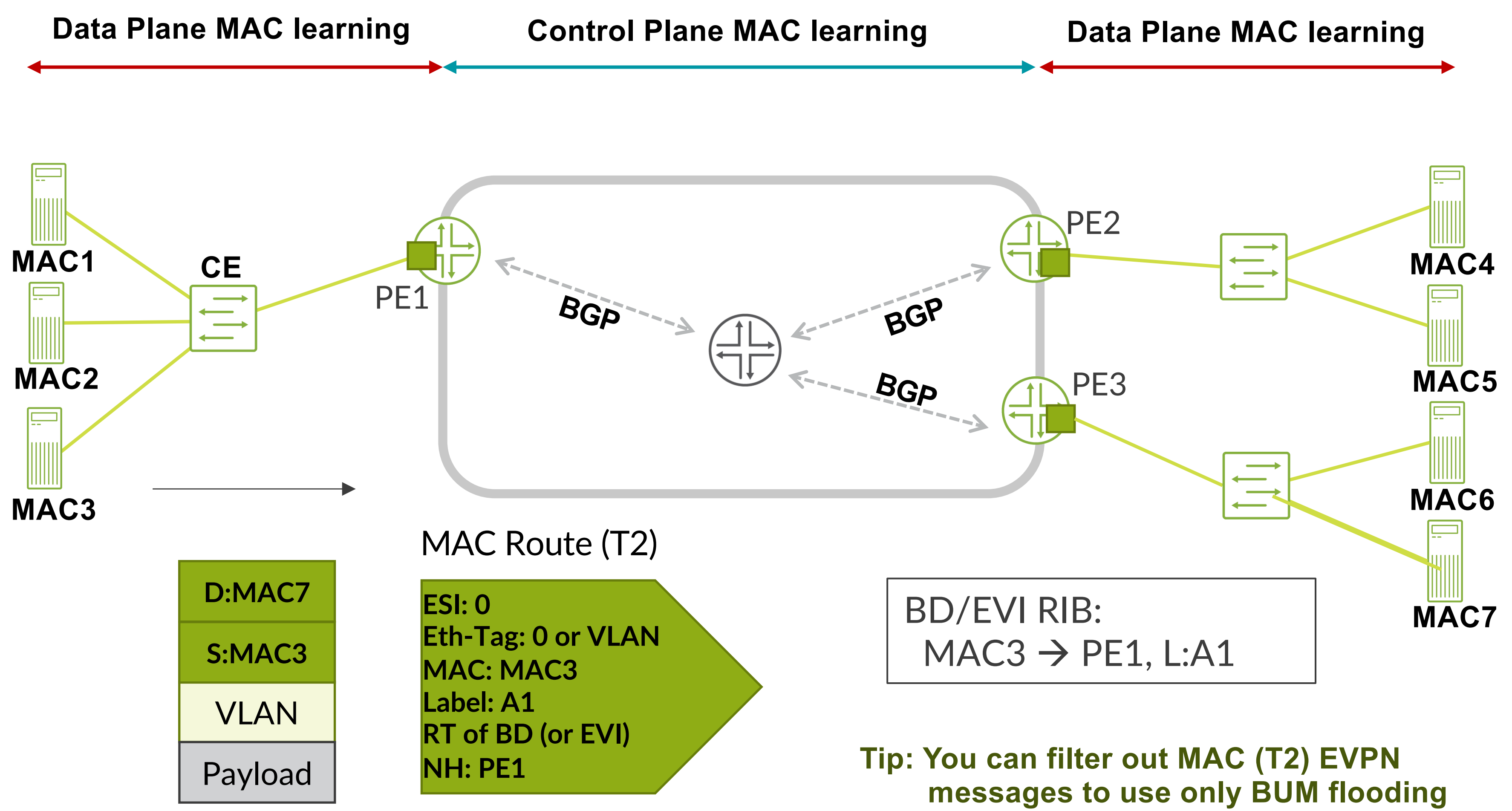


Tip: You can filter out IMET (T3) EVPN messages to prevent BUM flooding

PE1 performs ingress replication and sends each replicated copy of BUM frame via P2P MPLS tunnel to each remote PE that advertised IMET (T3) for given BD (or EVI).

PE2/PE3 locally floods the frame (received with previously advertised flooding label: F2/F3)

EVPN OPERATION: MAC LEARNING

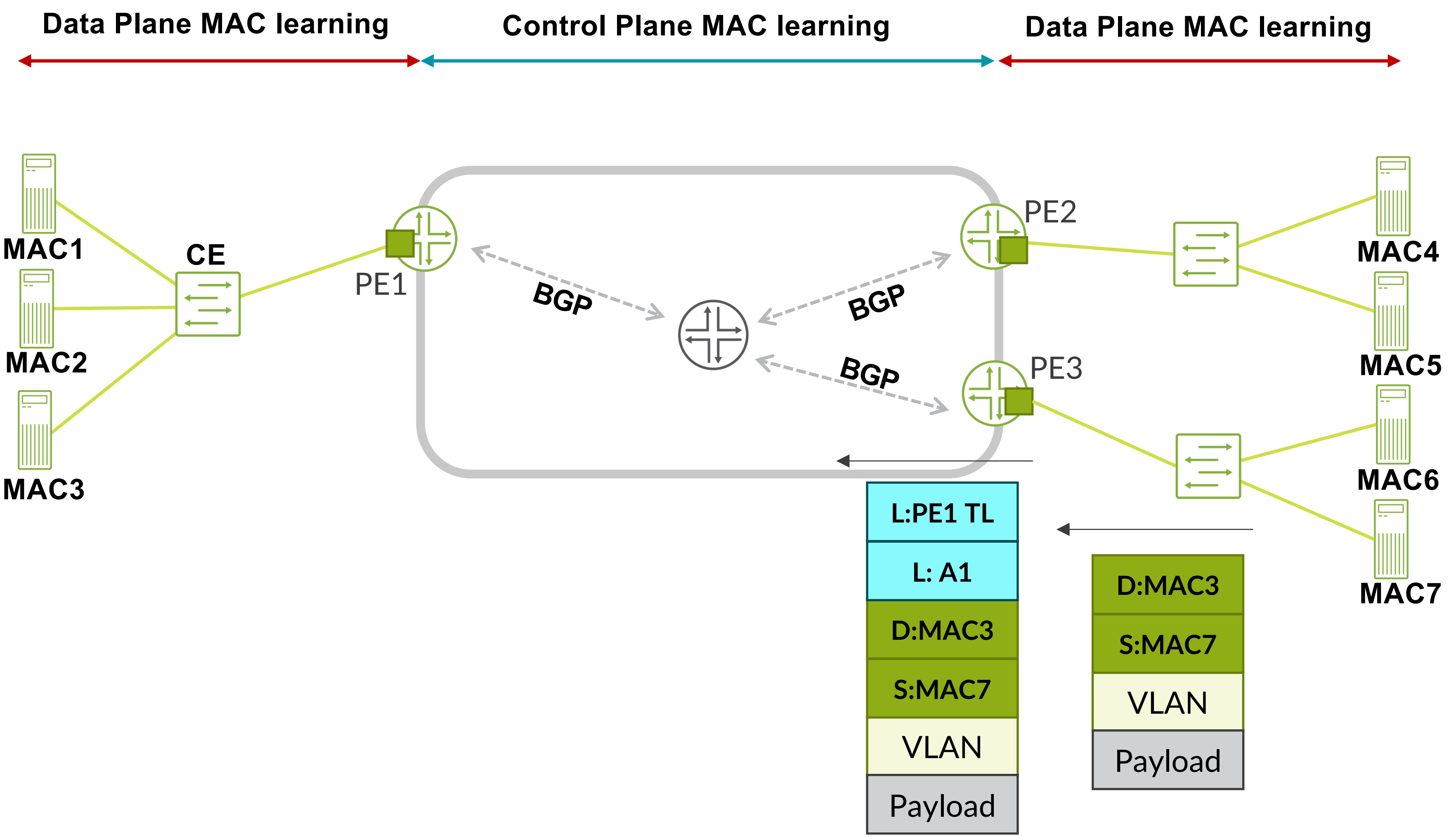


Tip: You can filter out MAC (T2) EVPN messages to use only BUM flooding

Data plane MAC learning of frames received over local interface

Control plane MAC learning (via MAC – T2 – messages) for frames received from remote PEs

EVPN OPERATION: KNOWN UNICAST FORWARDING



Frame forwarded by PE3 to PE1 via P2P MPLS tunnel to PE1

PE1 performs label lookup (A1) to determine EVI

For VLAN-aware bundle only, PE1 performs VLAN lookup to determine BD

PE1 performs MAC lookup to determine local interface

EVPN

Multi-homing

ETHERNET SEGMENT IDENTIFIER (ESI)

Used for EVPN Multi-Homing

Type	Semantics
0x00	An arbitrary nine-octet value, configured by operator
0x01	When IEEE 802.1AX LACP used between the PEs and CEs, an auto-generated ESI value determined from LACP System ID
0x02	Used when indirectly connected hosts via a bridged LAN between the CEs and the PEs. Auto-generated ESI derived from Root Bridge ID
0x03	MAC-based ESI that can be auto-generated or configured
0x04	Router-ID based Value; auto-generated or configured.
0x05	AS-based value; auto-generated or configured

A 10 octet value with range from 0x00 to 0xFFFFFFFFFFFFFFFFFFFF

- First octet denotes ESI type
- 9 octets might be used for ESI differentiation

For a single-homed CE, ESI value is 0

When CE is multi-homed, non 0 ESI must be used

- Must be unique across the entire network
- Per-IFD (ESI shared by multiple EVIs) or per-IFL (ESI unique to EVI) ESI is supported

EVPN DESIGNATED FORWARDER ELECTION ON MH ESI

EVPN Designated Forwarder (DF) responsibility

- Only DF sends BUM traffic towards CE (valid for both single-active and all-active MH)
- In single-active MH, IFLs in non-DF state are blocked (both input/output direction)

EVPN DF election granularity

- Per <ESI, VLAN> or <ESI, VLAN-bundle> → i.e. per-IFL (RFC7432)
- Per <ESI> → i.e. per-IFD (valid with per-IFD allocation method only) (draft-brissette-bess-evpn-mh-pa)
- DF election happens based on information in exchanged Ethernet Segment (T4) routes → generated only for MH-ed ESI

EVPN DF election algorithm

- Service Carving (RFC 7432)
  - Multiple VLANs (IFLs) on the same IFD (ESI), will have DF on different Pes, using modulus-based algorithm (PE = <ES, VLAN> mod N-PE)
  - Helps BUM load-balancing on per-VLAN (per-IFL) basis
- Manual, preference based (draft-rabadan-bess-evpn-pref-df)
- Emerging DF election methods
  - The Highest Random Weight DF Election Algorithm (RFC 8584)
  - The AC-Influenced DF Election (RFC 8584)
  - Fast Recovery for EVPN DF Election (draft-ietf-bess-evpn-fast-df-recovery)
  - Per multicast flow Designated Forwarder Election for EVPN (draft-ietf-bess-evpn-per-mcast-flow-df-election)
  - Weighted Multi-Path Procedures for EVPN All-Active Multi-Homing (draft-ietf-bess-evpn-unequal-lb)

EVPN DESIGNATED FORWARDER ELECTION ON MH ESI

ES (T4) Route doesn't belong to any specific EVI

ESI (9 significant octets)

```
root@R1> show route advertising-protocol bgp 192.168.0.4 match-prefix 4:* detail table
__default_evpn__.evpn.0

__default_evpn__.evpn.0: 6 destinations, 6 routes (6 active, 0 holddown, 0 hidden)
* 4:9.9.9.9:0::112233445566778899:9.9.9.9/296 ES (1 entry, 1 announced)
 BGP group IBGP-TO-RR type Internal
     Route Distinguisher: 9.9.9.9:0
     Nexthop: 9.9.9.9
     Localpref: 100
     AS path: [65000] I
     Communities: es-import-target:11-22-33-44-55-66
```

```
root@R1> show route advertising-protocol bgp 192.168.0.4 table bgp.rtarget.0
         match-prefix 65000:11*

bgp.rtarget.0: 32 destinations, 32 routes (32 active, 0 holddown, 0 hidden)
  Prefix                  Nexthop              MED     Lclpref    AS path
  65000:11-22-33-44-55-66/96
*                         Self                         100        I
```

No "classical" RT community

New **ES-Import-Target** extended community, not associated with any EVI, but derived automatically from ESI (6 octets only, due to extended community size limit)

RT-constraints automatically generated, restricting the distribution of the ES (T4) route to only PEs connected to the same Ethernet Segment

EVPN DESIGNATED FORWARDER ELECTION ON MH ESI

Assume following ESIs:

- ESI = 00:00:00:00:00:00:00:00:00:01 → PE1 + PE2
- ESI = 00:00:00:00:00:00:00:00:00:02 → PE2 + PE3
- ESI = 00:00:00:00:00:00:00:00:00:03 → PE3 + PE4
- ESI = 00:00:00:00:00:00:00:00:00:04 → PE4 + PE1
- …
- ESI = 00:00:00:00:00:00:00:FF:FF:FF → PEx + PEy

Generated ES-Import-Target extended community (and RT-constraints) for all above ESIs:

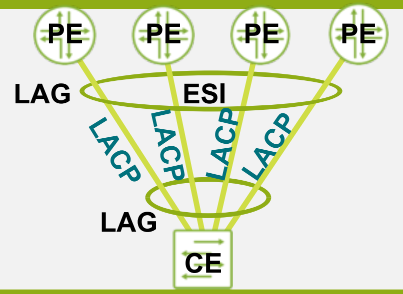
- 00:00:00:00:00:00 → first 6 octets following ESI Type octet

ES (T4) routes distributed across all PEs, not only to PEs connected to specific ES → very inefficient

Tip: Differentiate ESIs within 6 octets following ESI Type.
Leave last 3 octets set to 00:00:00.

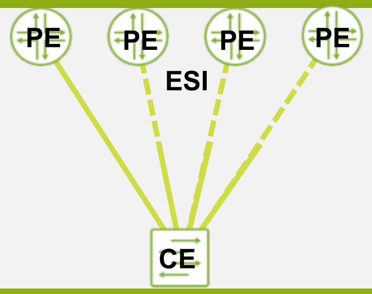
EVPN MULTI-HOMING (MH) OPTIONS

EVPN ALL-ACTIVE MH



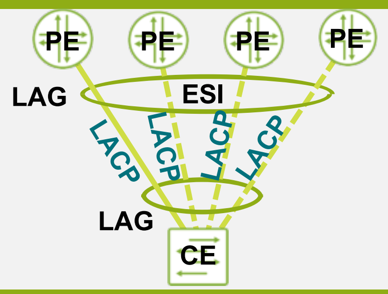
- à la MC-LAG A/A
- PE side:
  - AE or native IFD (typically AE)
  - standby (nDF) IFLs in 'up' state
- CE side:
  - AE IFD
  - standby IFLs: none
- LACP: optional
- DF election: any method
- Known unicast: all-active load-balancing
- BUM:
  - PE→CE: single-active forwarding
  - CE→PE: all-active load-balancing

EVPN SINGLE-ACTIVE MH (PER-IFL)



- à la VPLS A/S MH
- PE side:
  - AE or native IFD (typically native)
  - standby (nDF) IFLs in 'ccc-down' state
- CE side:
  - native IFDs with IFLs in a BD
  - standby IFLs in 'up' state
- LACP: not used
- DF election: any method
- Known unicast: single-active forwarding
- BUM:
  - PE→CE: single-active forwarding
  - CE→PE: all-active flooding

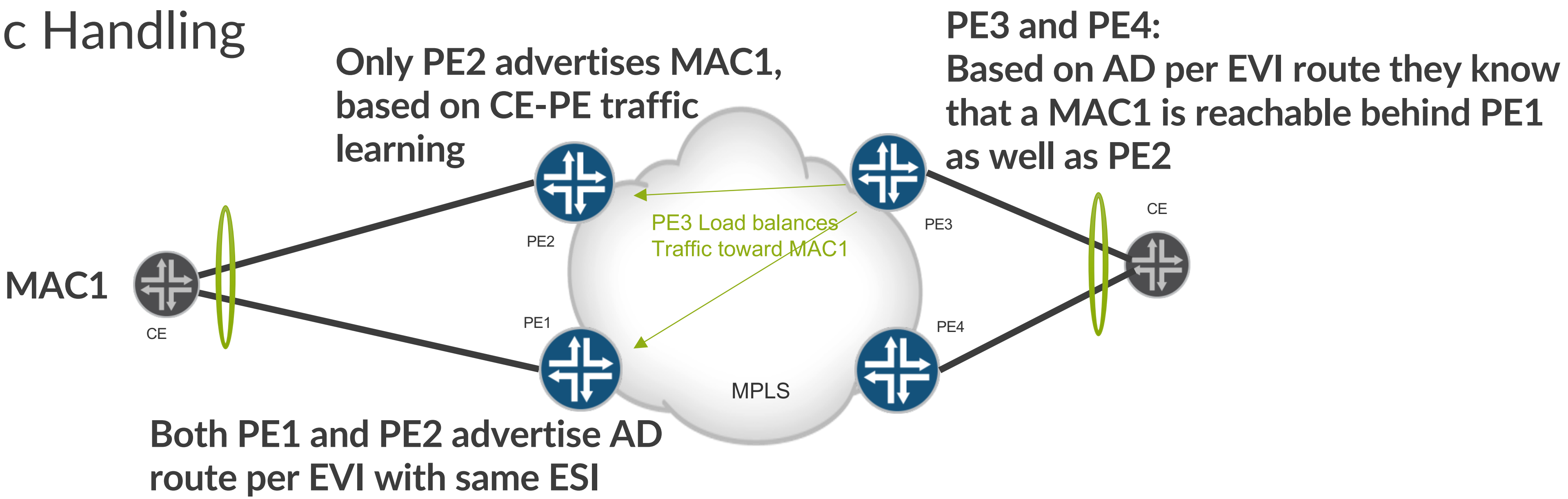
EVPN SINGLE-ACTIVE MH (PER-IFD)



- à la MC-LAG A/S
- PE side:
  - AE IFD
  - standby (nDF) IFLs in 'ccc-down' state
- CE side:
  - AE IFD
  - standby IFLs: none
- LACP: mandatory (to put down standby IFD)
- DF election: per IFD only
- Known unicast: single-active forwarding
- BUM:
  - PE→CE: single-active forwarding
  - CE→PE: single-active forwarding

EVPN ALL-ACTIVE MULTI-HOMING (ALIASING)

Unicast Traffic Handling



Traffic from CE is load balanced across PEs

Aliasing – load balancing of traffic to CE

- Remote PEs load-balance traffic across PEs advertising the same ESI
- Even when the MAC address is learned only by one PE
- Remote PE uses aliasing label to send traffic to a PE not advertising the MAC address

EVPN ALL-ACTIVE MULTI-HOMING (ALIASING)

Ethernet Auto-Discovery (T1) Route per-EVI (Eth-Tag-ID ≠ 0xFFFF:FFFF)

ESI (9 significant octets)

Eth-Tag ID

E A-D (T1) per-EVI Route belongs to some specific EVI

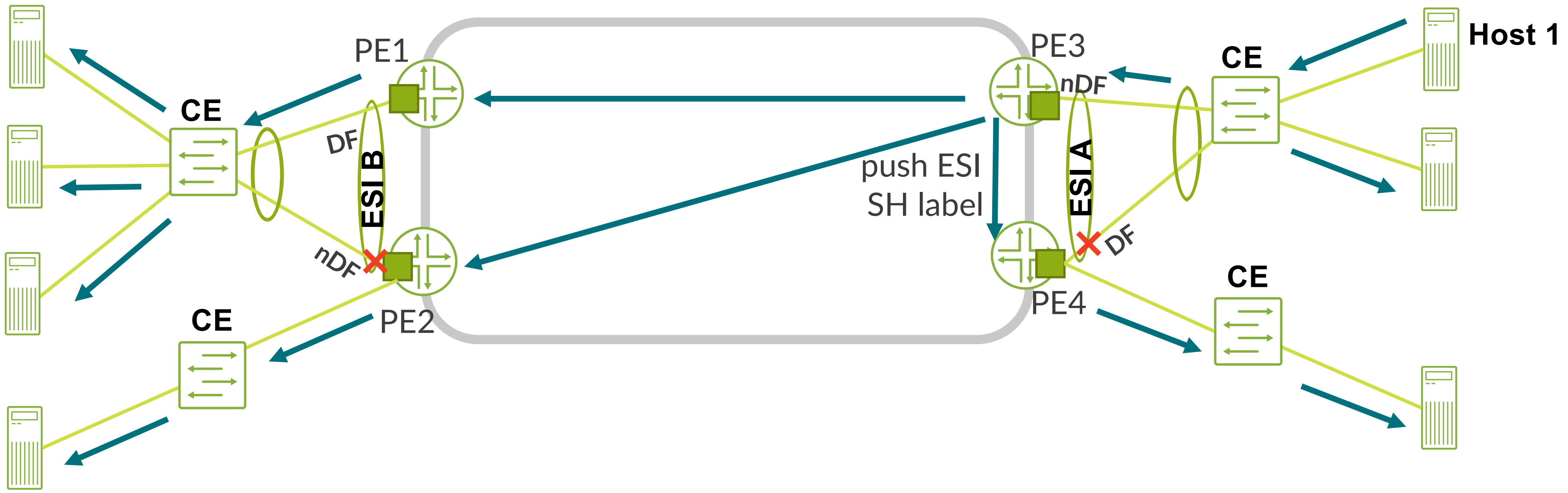
```
root@R1> show route advertising-protocol bgp 192.168.0.4 match-prefix 1:* table RI-EVPN-1 detail

RI-EVPN-1.evpn.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)
* 1:192.168.0.1:201::112233112233112233::0/192 AD/EVI (1 entry, 1 announced)
 BGP group IBGP-TO-RR type Internal
     Route Distinguisher: 192.168.0.1:201
     Route Label: 100070
     Nexthop: 9.9.9.9
     Localpref: 100
     AS path: [65000] I
     Communities: target:65000:1111
```

Aliasing label.

Classical RT extended community

EVPN ALL-ACTIVE MULTI-HOMING (SPLIT HORIZON)

BUM Traffic Handling



Split horizon filtering

- non-DF PE floods BUM traffic to DF with ESI split horizon (SH) label to identify source ES
- DF PE performs split horizon filtering and does not forward the traffic back to the CE

EVPN ALL-ACTIVE MULTI-HOMING (SPLIT HORIZON)

Ethernet Auto-Discovery (T1) Route per-ESI (Eth-Tag-ID = 0xFFFF:FFFF)

E A-D per-ESI (T1) Route generated from 'default' RIB

ESI

Ethernet Tag ID (0xFFFF:FFFF)

```
root@R2# run show route advertising-protocol bgp 192.168.0.4 match-prefix 1:*
         table __default_evpn__.evpn.0 detail

__default_evpn__.evpn.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
* 1:192.168.0.2:0::020122019900000::FFFF:FFFF/192 AD/ESI (1 entry, 1 announced)
 BGP group IBGP-TO-RR type Internal
     Route Distinguisher: 192.168.0.2:0
     Nexthop: Self
     Localpref: 100
     AS path: [65000] I
     Communities: target:65000:1111 target:65000:12323 target:65000:22301 target:65000:65199 esi-
label:0x0:all-active (label 79)
```

RD not specific to any EVI → it is not E A-D per-EVI (T1) route

ESI Split-Horizon label (the same for all EVIs sharing the ESI)

RTs of all EVIs sharing this ESI

EVPN ALL-ACTIVE MULTI-HOMING (SPLIT HORIZON)

Ethernet Auto-Discovery (T1) Route per-ESI (ESI = 0xFFFF:FFFF) functions

- Distribute ESI Split-Horizon label
  - Required only on PEs attached to the same ESI
- Mass withdrawal
  - Required on ingress PE (essentially → on all other PEs)
  - When multi-homed PE-CE link fails, egress PE withdraws corresponding E A-D (T1) route per-ESI
  - Ingress PE automatically invalidates all MAC (T2) routes associated with that ESI, announced previously by egress PE, so failover on ingress PE is faster

EVPN/VXLAN – SPLIT HORIZON USING IP TRANSPORT

RFC8365 says:

Since VXLAN and NVGRE encapsulations do not include the ESI label, other means of performing the split-horizon filtering function must be devised for these encapsulations. The following approach is recommended for split-horizon filtering when VXLAN (or NVGRE) encapsulation is used. Every NVE tracks the IP address(es) associated with the other NVE(s) with which it has shared multihomed ESs.

When the NVE receives a multi-destination frame from the overlay network, it examines the **source IP address in the tunnel header** (which corresponds to the ingress NVE) and filters out the frame on all local interfaces connected to ESs that are shared with the ingress NVE. With this approach, it is required that the ingress NVE perform replication locally to all directly attached Ethernet segments (regardless of the DF election state) for all flooded traffic ingress from the access interfaces (i.e., from the hosts). This approach is referred to as "Local Bias", and has the advantage that only a single IP address need be used per NVE for split-horizon filtering, as opposed to requiring an IP address per Ethernet segment per NVE.

EVPN VERSUS MC-LAG

Feature	EVPN	MC-LAG
More that 2 PEs in the multi-homing group		✗
Active/Standby support		
Active/Active support		
No Inter-chassis Link required		✗
Standardized, interoperable inter-chassis control protocol		✗
Integrated (no VRRP required) L3 gateway support		✗

ETHERNET VPN FORWARDING SUMMARY

Ethernet VPN Data Plane

- Similar to other Layer 2/3 VPN data plane
  - Outer label = LDP/RSVP label towards remote PE
  - Inner label = “EVPN” label (1 or more) learned from remote PE

Destination	Label 1 (Top)	Label 2	Label 3 (Bottom)
Unicast single-homed	RSVP/LDP	MAC/IP	N/A
Unicast single-active	RSVP/LDP	MAC/IP	N/A
Unicast all-active	RSVP/LDP	MAC/IP Aliasing	N/A
BUM single-homed	RSVP/LDP	Inclusive Multicast	
BUM single-active	RSVP/LDP	Inclusive Multicast	Split Horizon
BUM all-active	RSVP/LDP	Inclusive Multicast	Split Horizon

Destination	VXLAN
Unicast single-homed	VNI
Unicast single-active	VNI
Unicast all-active	VNI
BUM single-homed	VNI
BUM single-active	VNI Source IP address
BUM all-active	VNI Source IP address

EVPN

IP Routing (Type 5)

IP PREFIX ROUTE – EVPN ROUTE-TYPE 5

EVPN Type 5 route is used to inter-subnet connectivity, exchange and install IP Prefix information between Tenants.

Standard: draft-rabadan-l2vpn-evpn-prefix-advertisement

Initial use cases:

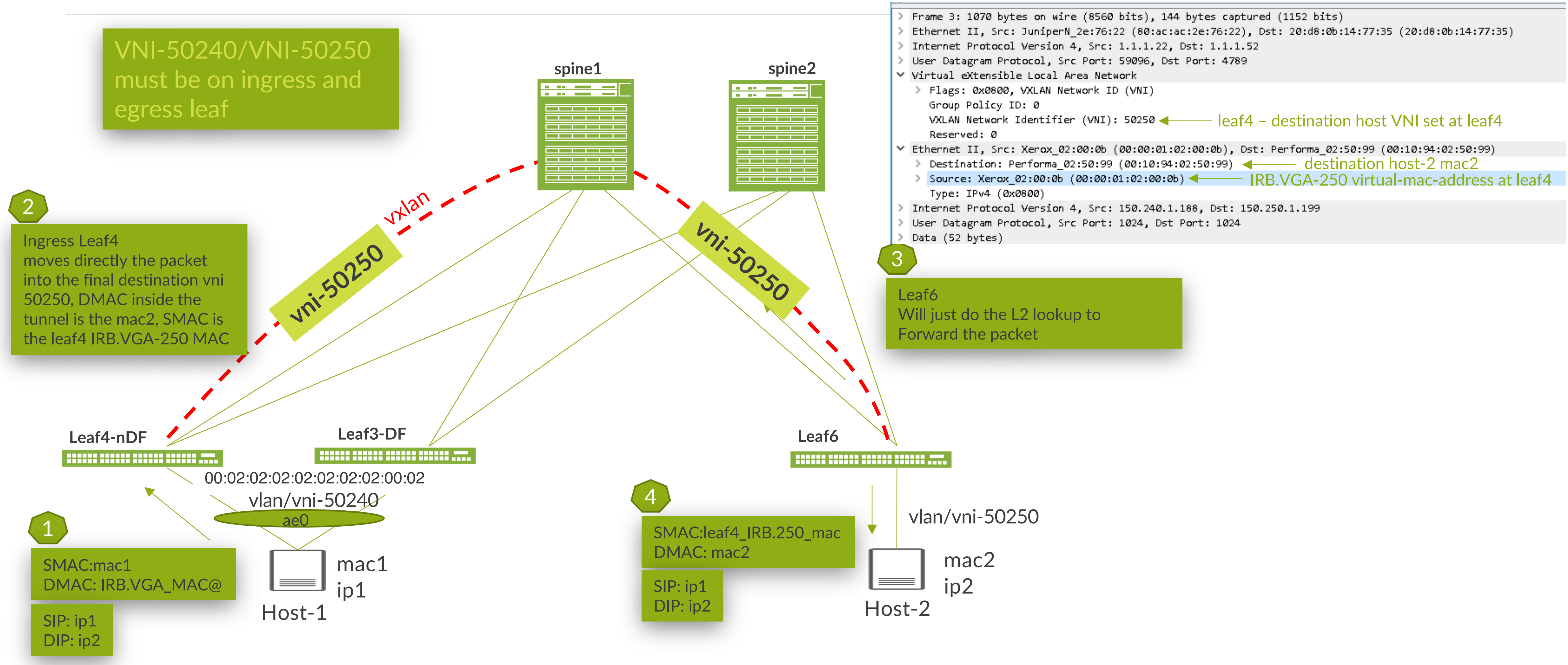
- IP Routing between Tenants
  - Asymmetric and Symmetric Routing
- DCI (Datacenter Interconnection)
  - No L2 stretch required between DCs
  - MACs belonging to a DC customer can be summarized by an IP prefix
  - End to end unified EVPN Solution. Use EVPN for the DC as well as DCI

```
+---------------------------------------+
|      RD    (8 octets)                 |
+---------------------------------------+
|Ethernet Segment Identifier (10 octets)|
+---------------------------------------+
|  Ethernet Tag ID (4 octets)           |
+---------------------------------------+
|  IP Prefix Length (1 octet)           |
+---------------------------------------+
|  IP Prefix (4 or 16 octets)           |
+---------------------------------------+
|  GW IP Address (4 or 16 octets)       |
+---------------------------------------+
|  MPLS Label (3 octets)                |
+---------------------------------------+
```

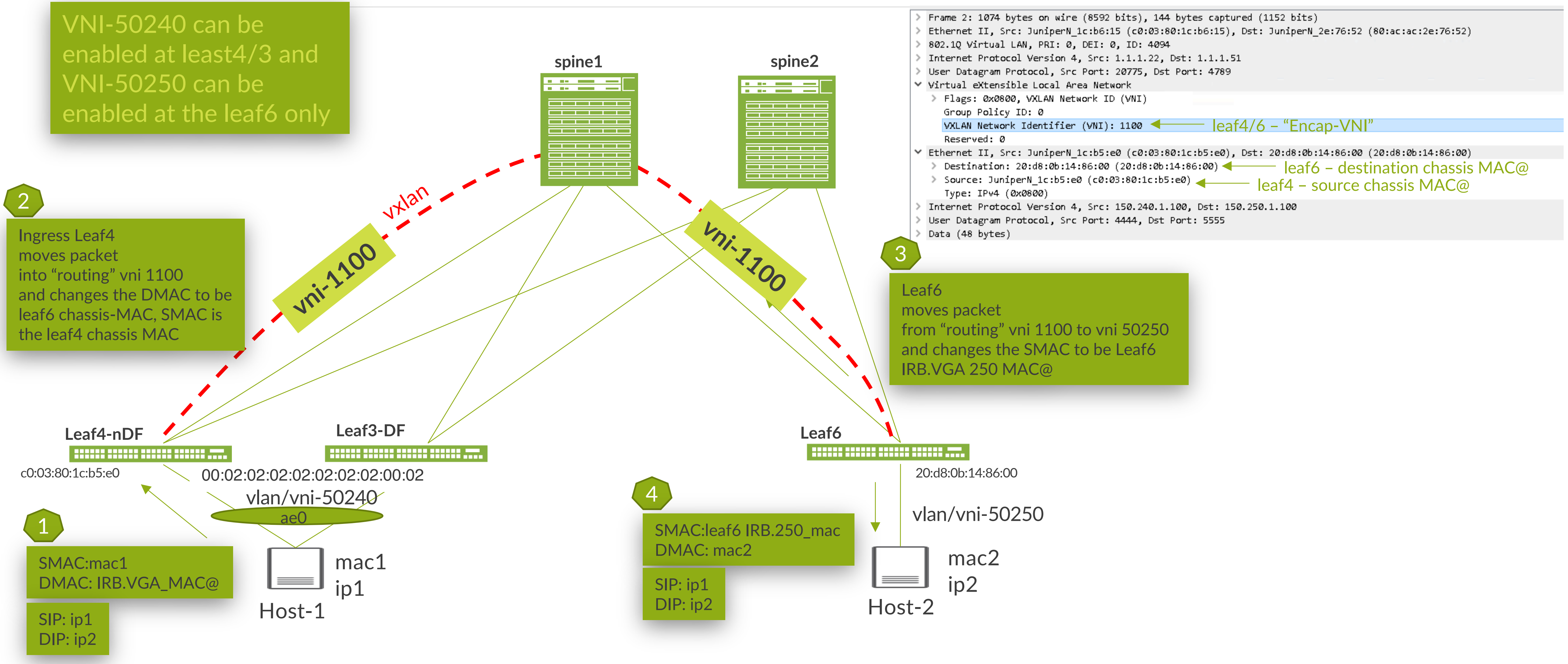
Carry following Extended Communities:

- L3 VNI RT (Auto-derivation from L3 VNI)
- Router's MAC Extended Community
- Encapsulation VXLAN

EVPN-VXLAN – ASYMMETRIC ROUTING – USING T2 ROUTES



EVPN-VXLAN – SYMMETRIC ROUTING USING TYPE-5 INSTANCE



EVPN

BUM optimization

ARP flooding reduction

BROADCAST FLOODING

Large broadcast flooding (e.g. ARP) might negatively impact DC operation

- 600k hosts with 10 min ARP cache timeout → average 1k pps of ARP Requests
- Routers connected to DC might need to process large number of ARPs
  - Typically, it happens in “slow path” (software processing)
  - Can cause heavy load on the router’s CPU
  - Typically limitation are low thousands per second

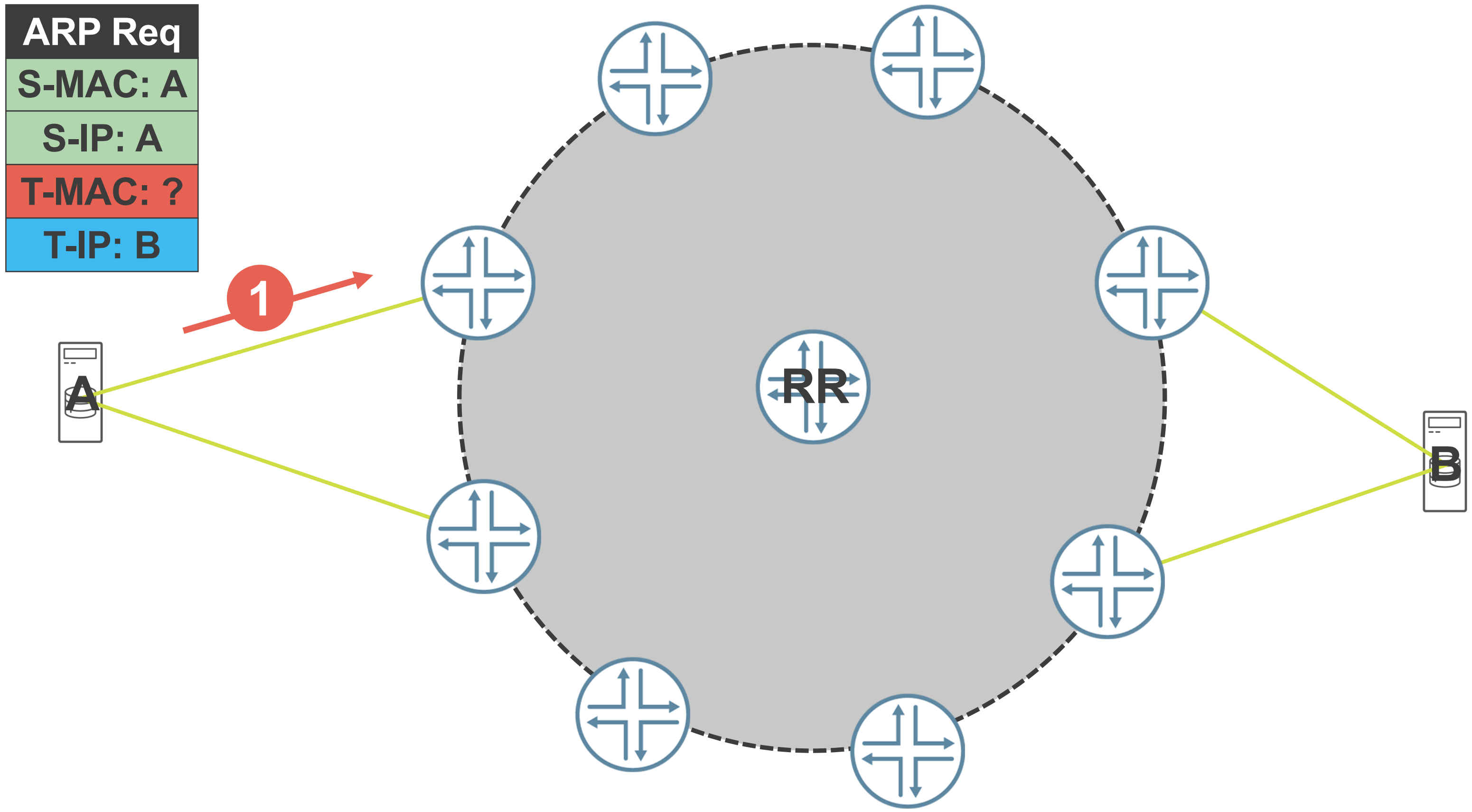
Historically, some attempts have been made to address the problem:

- RFC 6820: Address Resolution Problems in Large Data Center Networks

EVPN brings holistic way to suppress ARP storms

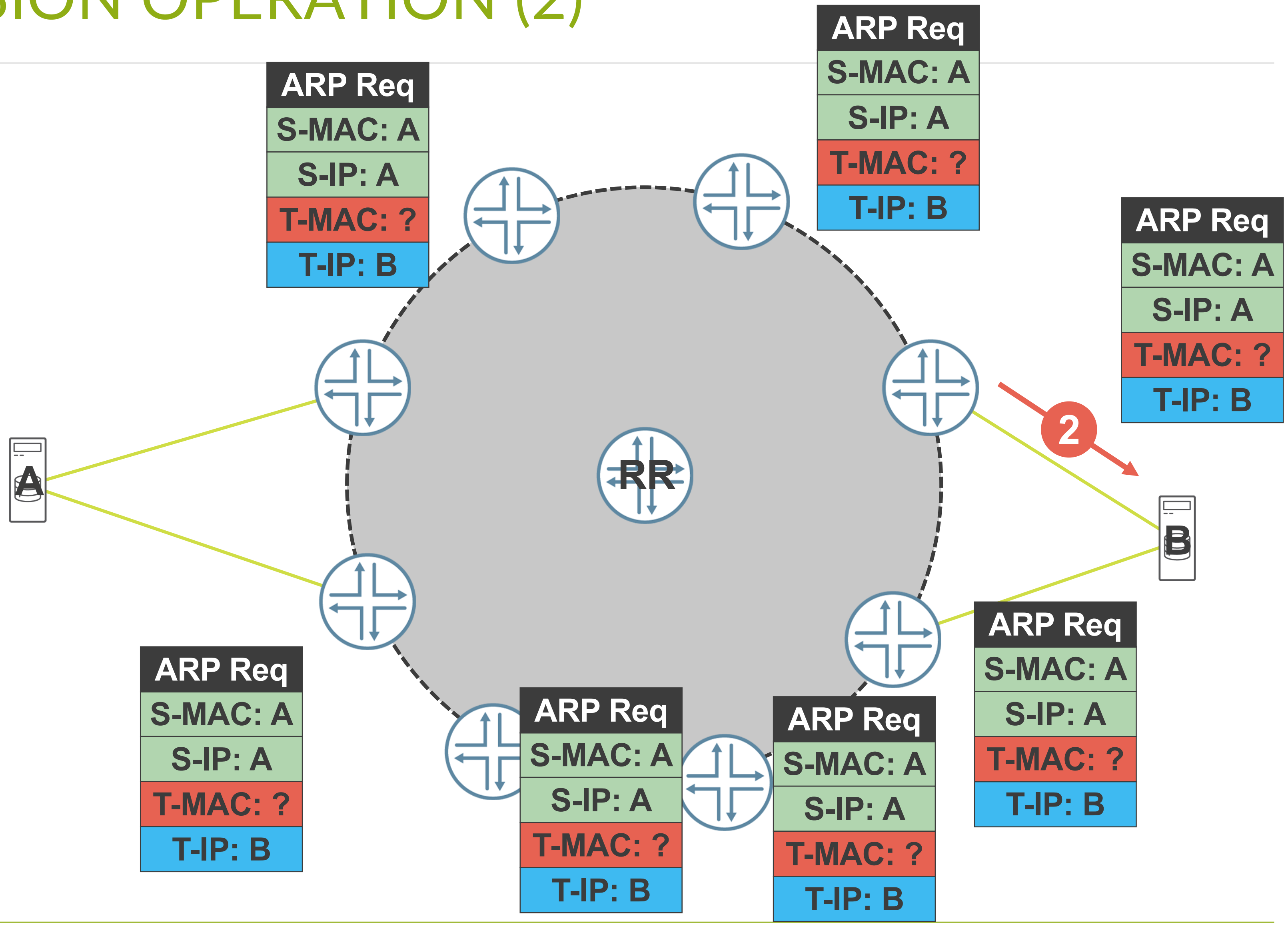
EVPN ARP SUPPRESSION OPERATION (1)

Host ’A’ issues ARP Request to resolve IP address ‘B’



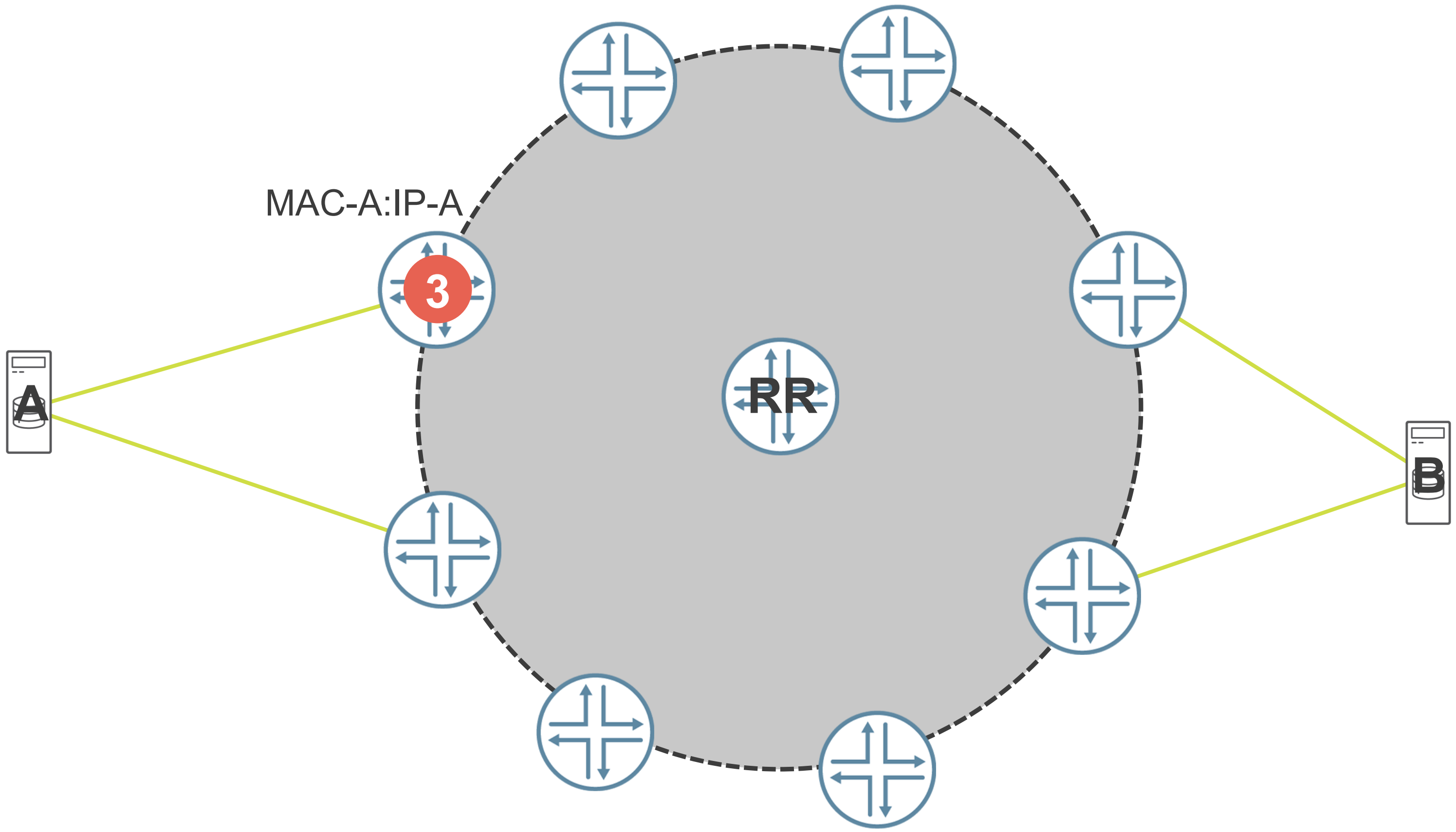
EVPN ARP SUPPRESSION OPERATION (2)

EVPN PE router, where ARP Request (with broadcast D-MAC) arrives, floods its via EVPN machinery, eventually arriving to host B



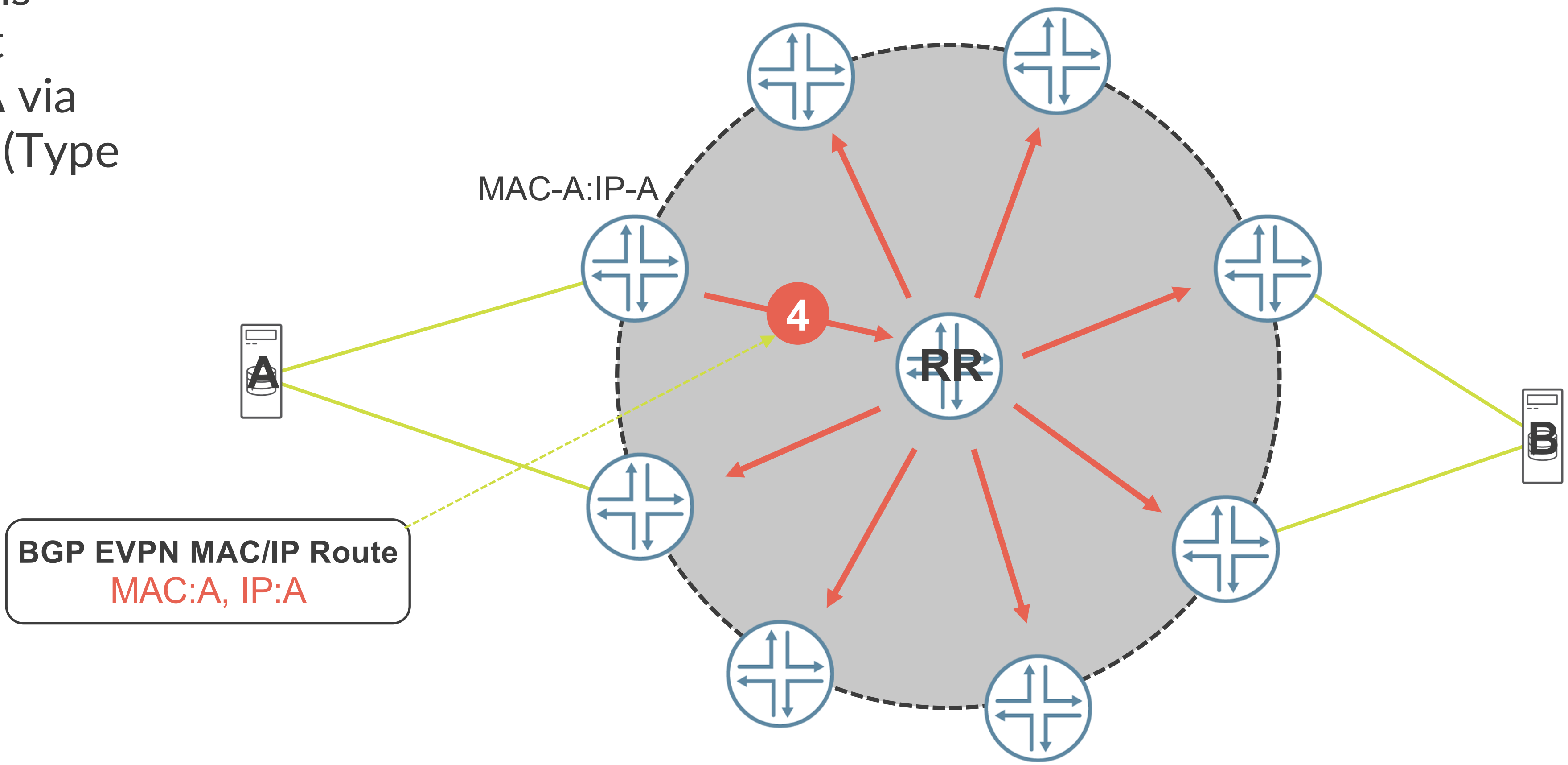
EVPN ARP SUPPRESSION OPERATION (3)

In the meantime, ingress EVPN PE intercepts ARP Request, learns MAC-A:IP-A association from it, and updates its EVPN database



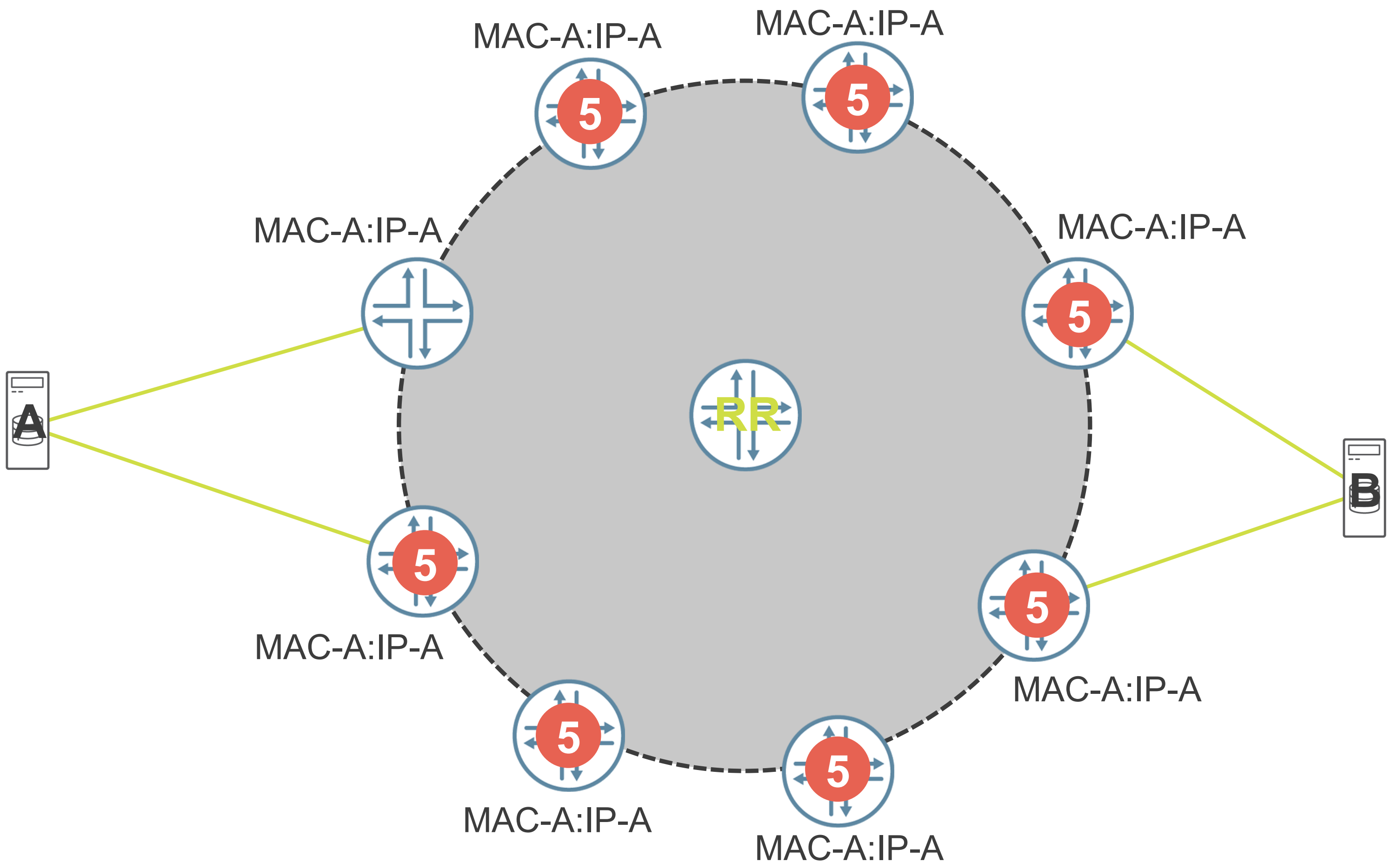
EVPN ARP SUPPRESSION OPERATION (4)

Ingress EVPN informs remaining PEs about learned MAC-A:IP-A via BGP EVPN MAC/IP (Type 2) Route



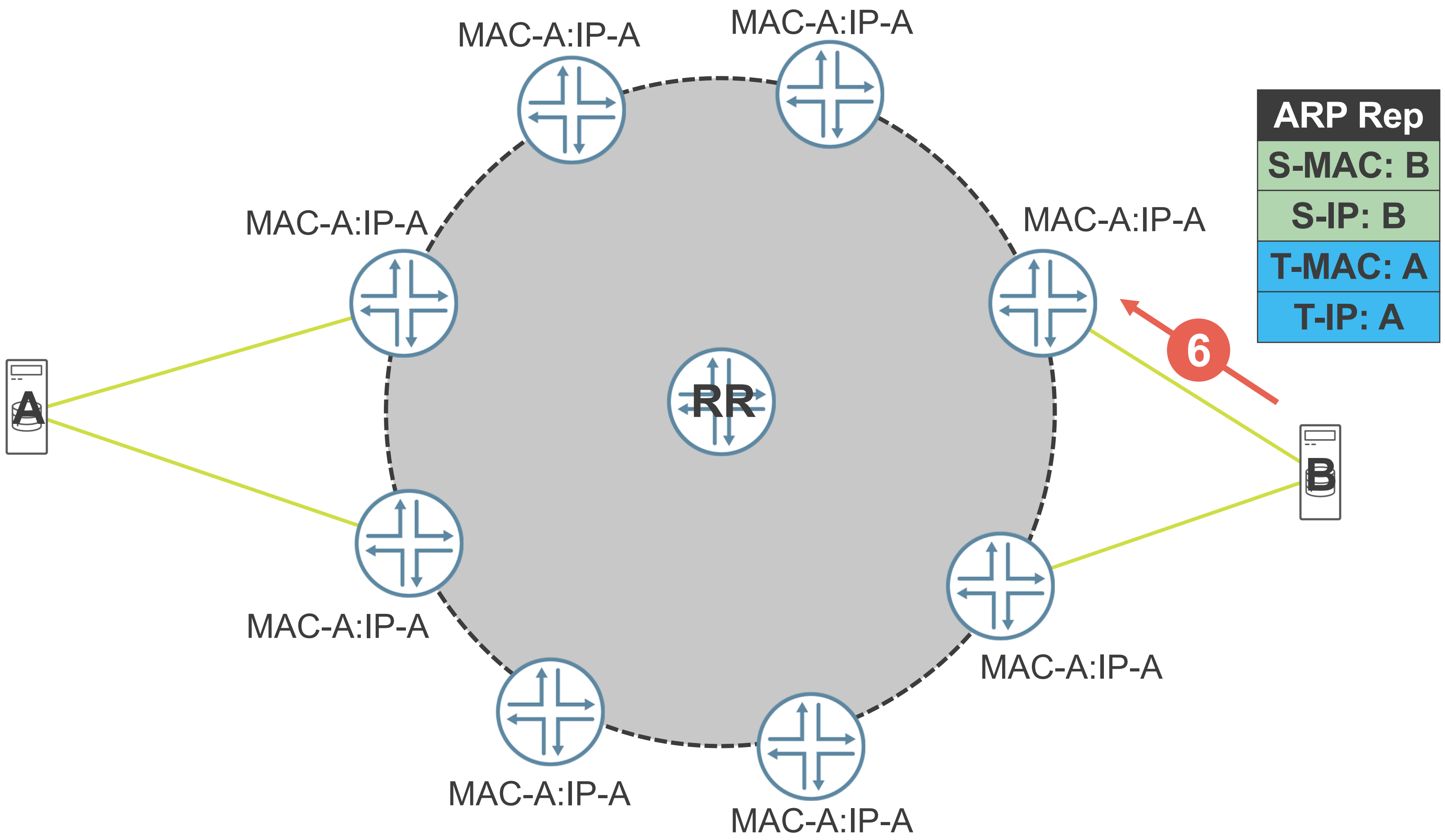
EVPN ARP SUPPRESSION OPERATION (5)

Remaining EVPN PEs update their EVPN database with MAC-A:IP-A association learned from ingress PE. Eventually, all PEs know about MAC-A:IP-A



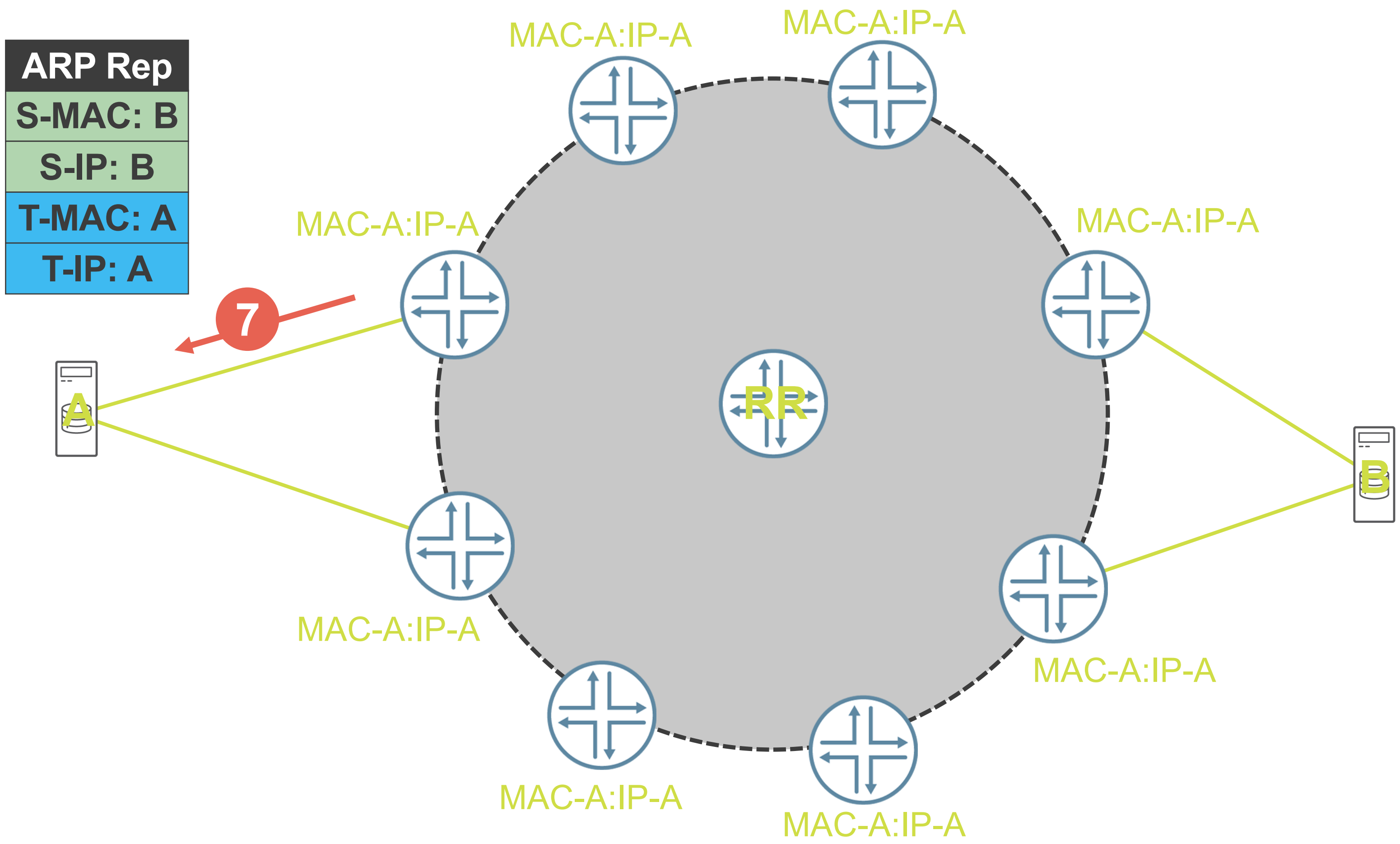
EVPN ARP SUPPRESSION OPERATION (6)

Host-B answers with ARP Reply



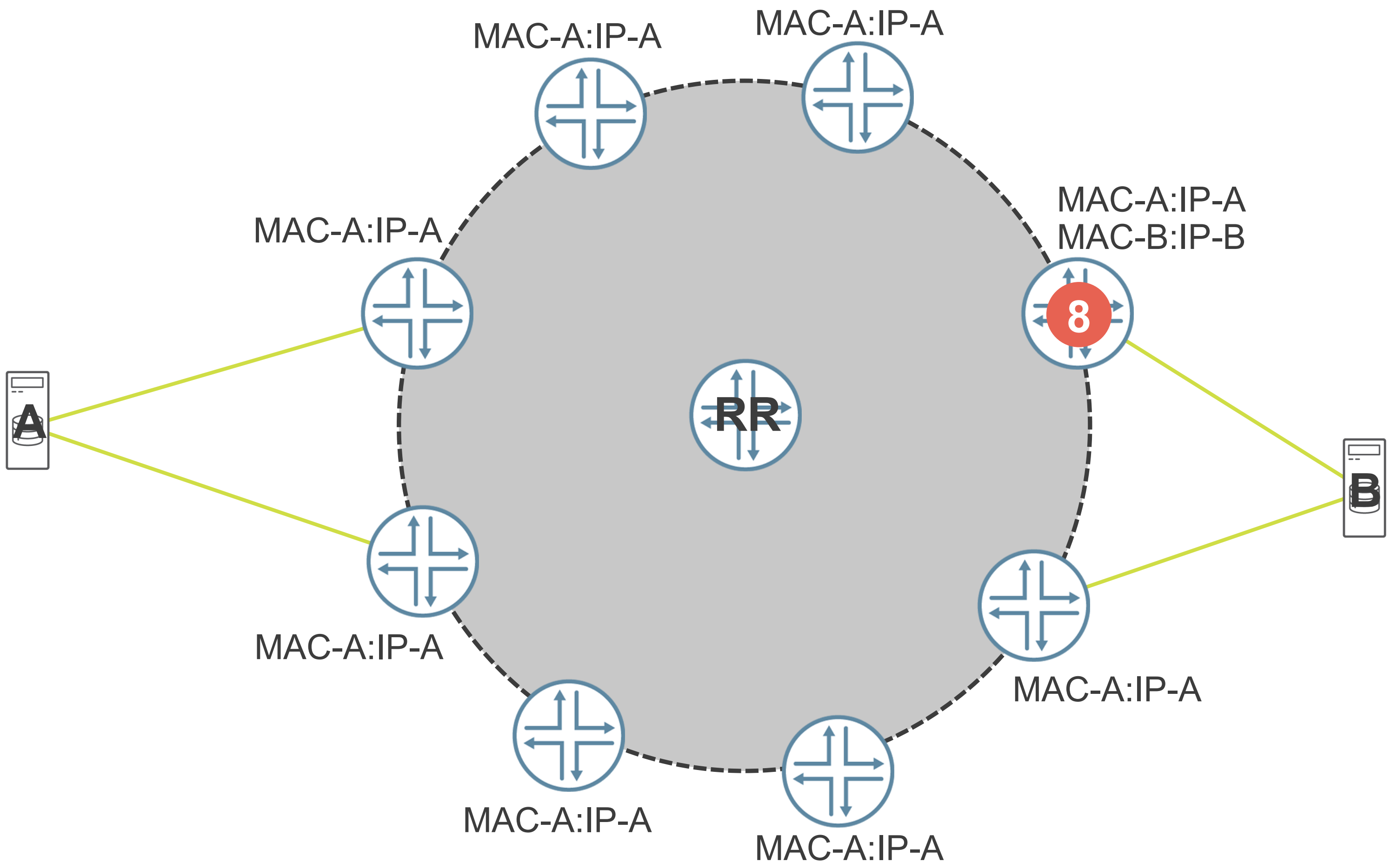
EVPN ARP SUPPRESSION OPERATION (7)

EVPN PE router, where ARP Reply arrives, has already MAC-A entry in its EVPN database, so ARP Reply is unicasted (not broadcasted) via EVPN machinery, and eventually arrives at Host-A



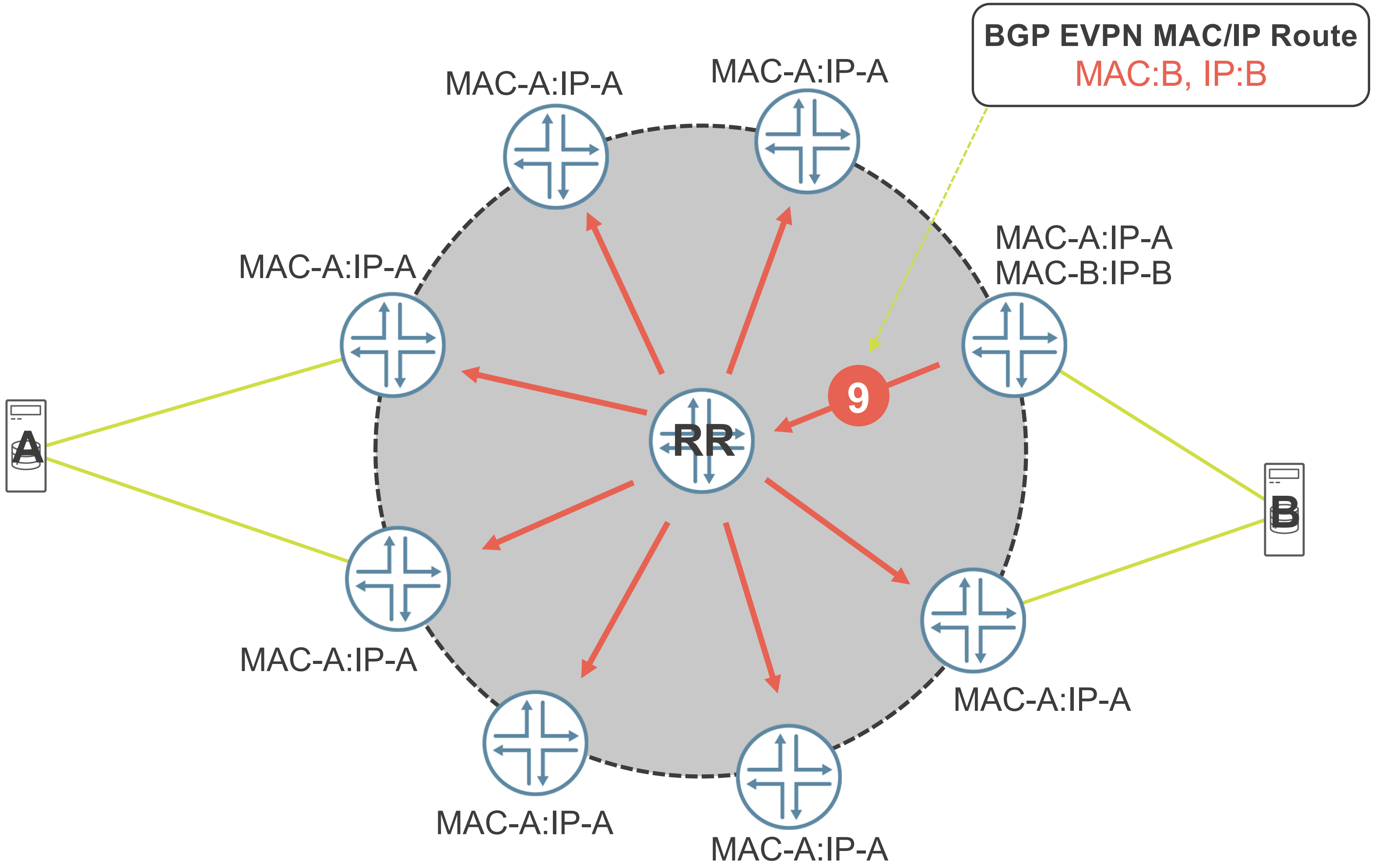
EVPN ARP SUPPRESSION OPERATION (8)

In the mean time, EVPN PE intercepts ARP Reply, learns MAC-B:IP-B association from it, and updates its EVPN database



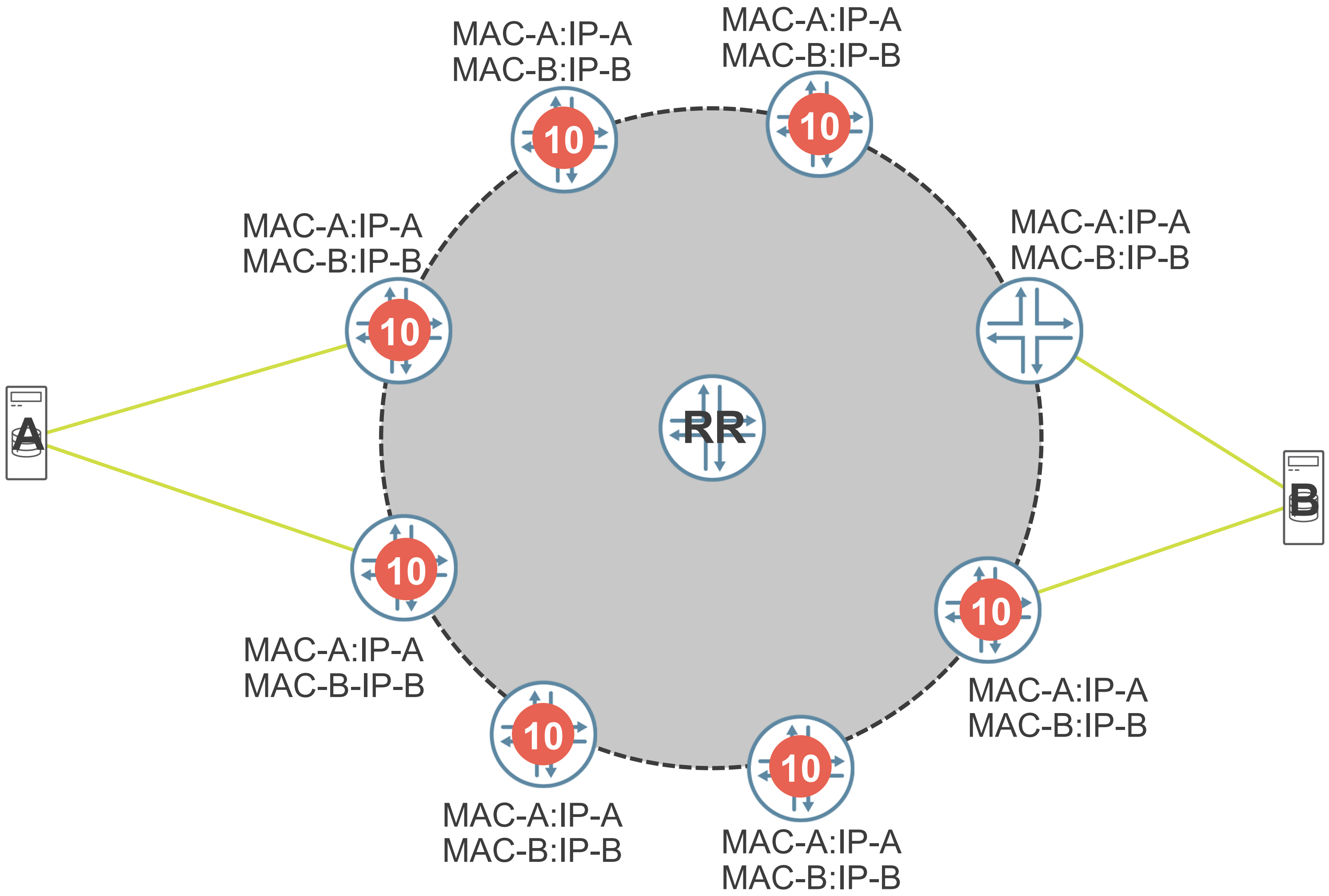
EVPN ARP SUPPRESSION OPERATION (9)

Ingress EVPN informs remaining PEs about learned MAC-B:IP-B via BGP EVPN MAC/IP (Type 2) Route



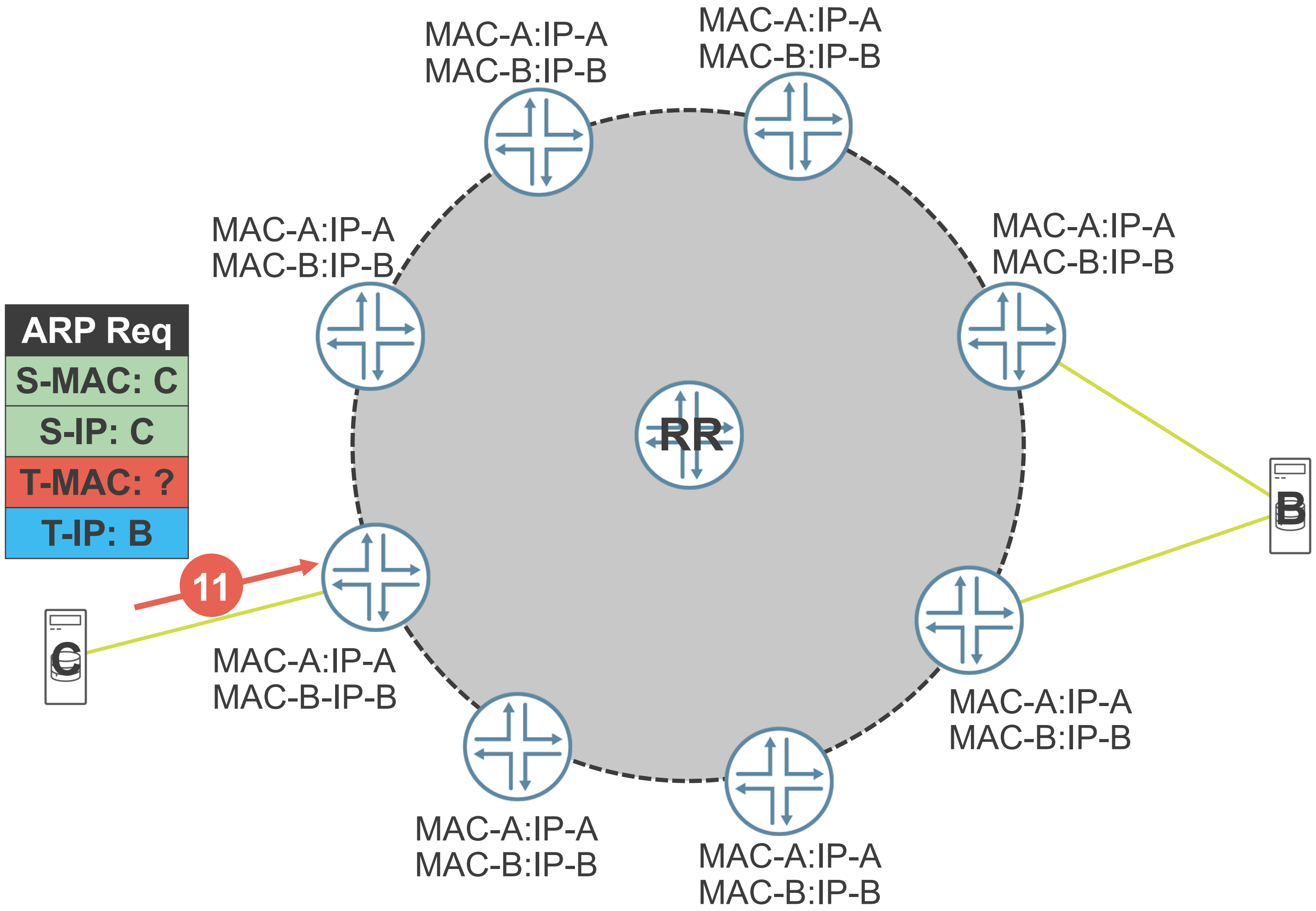
EVPN ARP SUPPRESSION OPERATION (10)

Remaining EVPN PEs update their EVPN database with MAC-B:IP-B association learned from ingress PE. Eventually, all PEs know about MAC-A:IP-A and MAC-B:IP-B



EVPN ARP SUPPRESSION OPERATION (11)

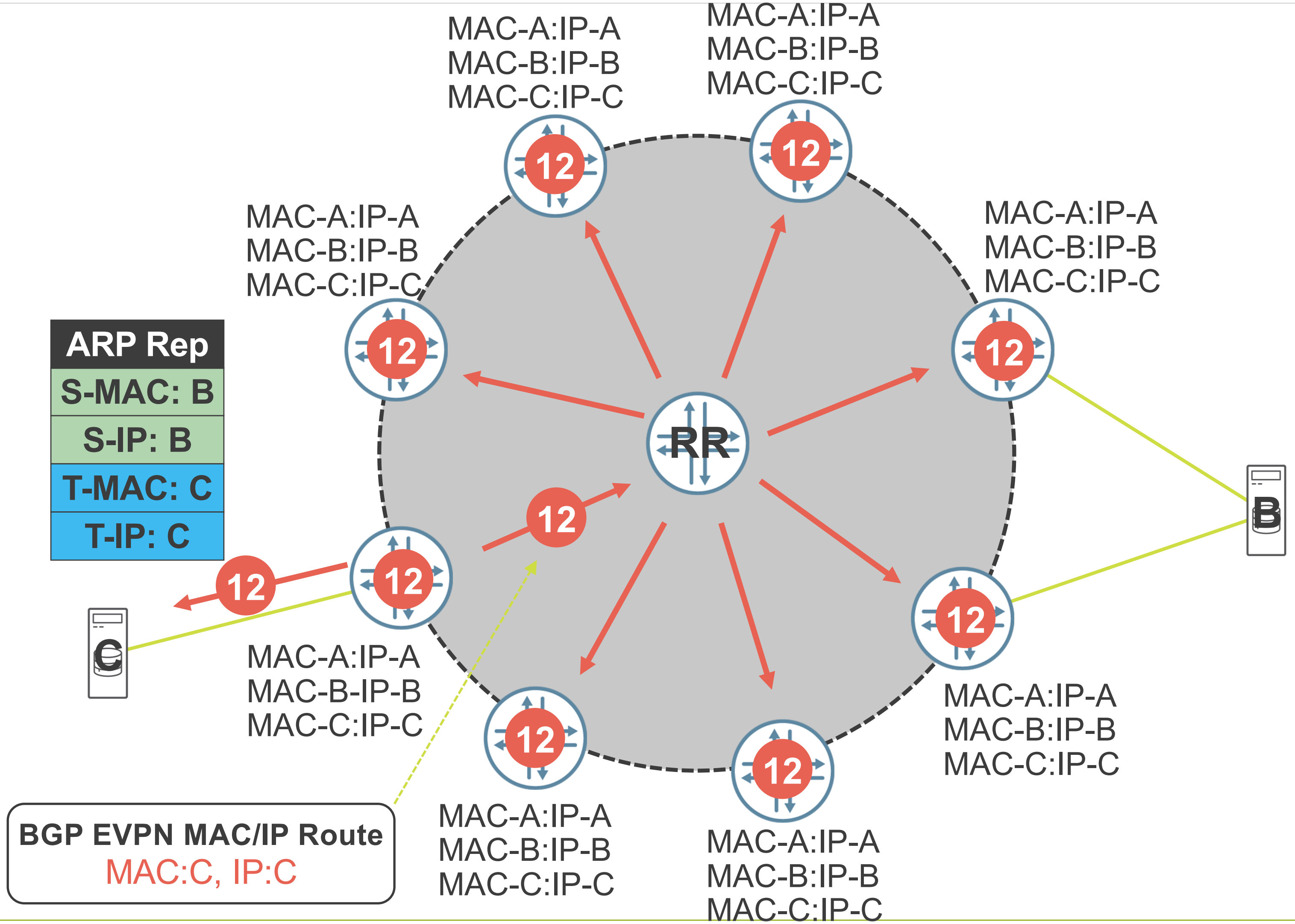
Host ’C’ issues ARP Request to resolve IP address ‘B’



EVPN ARP SUPPRESSION OPERATION (12)

EVPN PE already has an entry for MAC-B:IP-B, so it

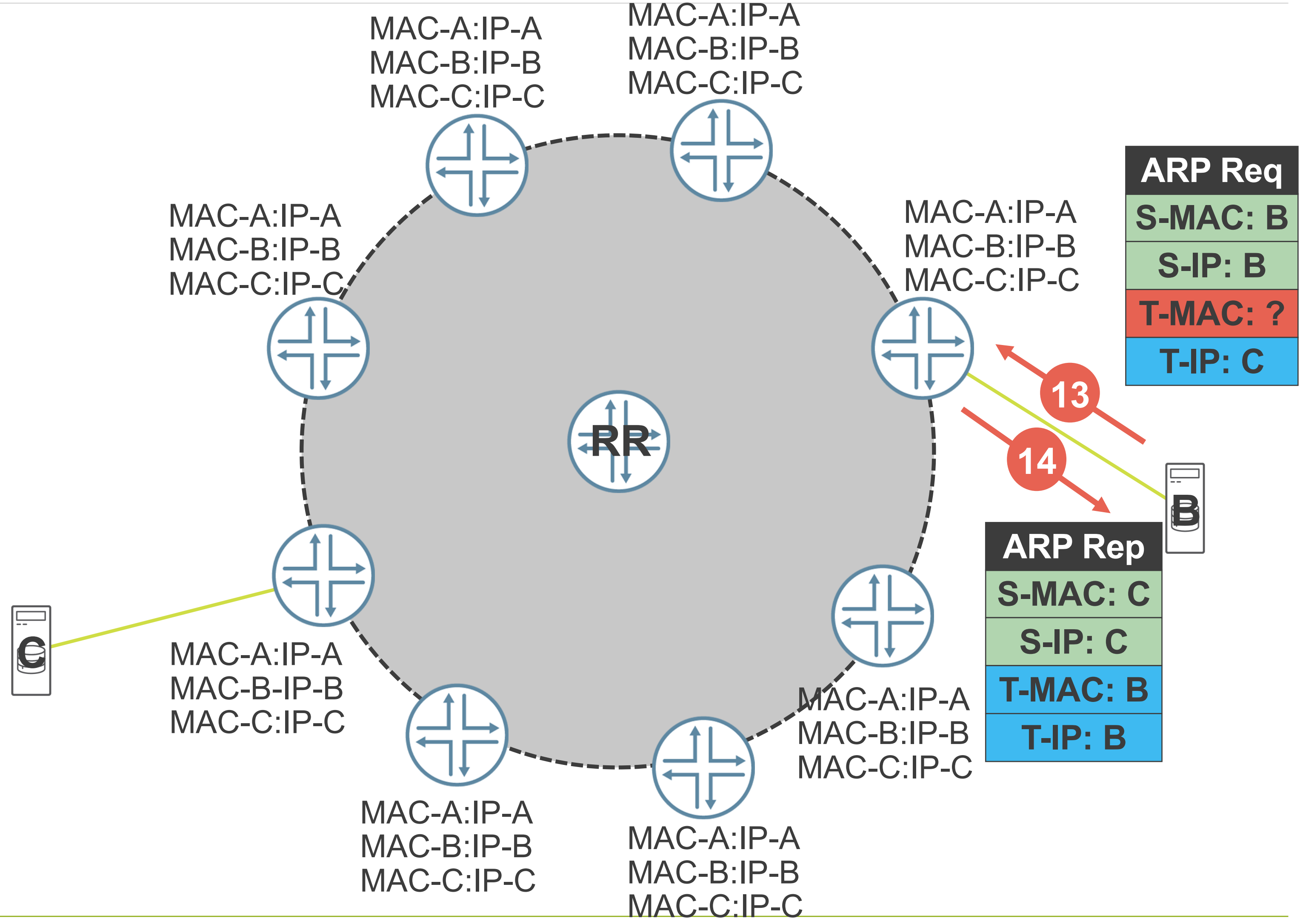
- sends ARP Reply to host C
- Learns MAC-C:IP-C
- Informs remaining PEs about MAC-C:IP-C



EVPN ARP SUPPRESSION OPERATION (13, 14)

When ARP cache on Host-B expires, Host-B issues ARP Request

- suppressed on PE
- PE sends immediate ARP Reply
- No update in EVPN BGP machinery required



EVPN ND SUPPRESSION

ND suppression follows similar concepts to ARP suppression, hence not discussed explicitly in this session

EVPN
BUM optimization
Multicast flooding reduction

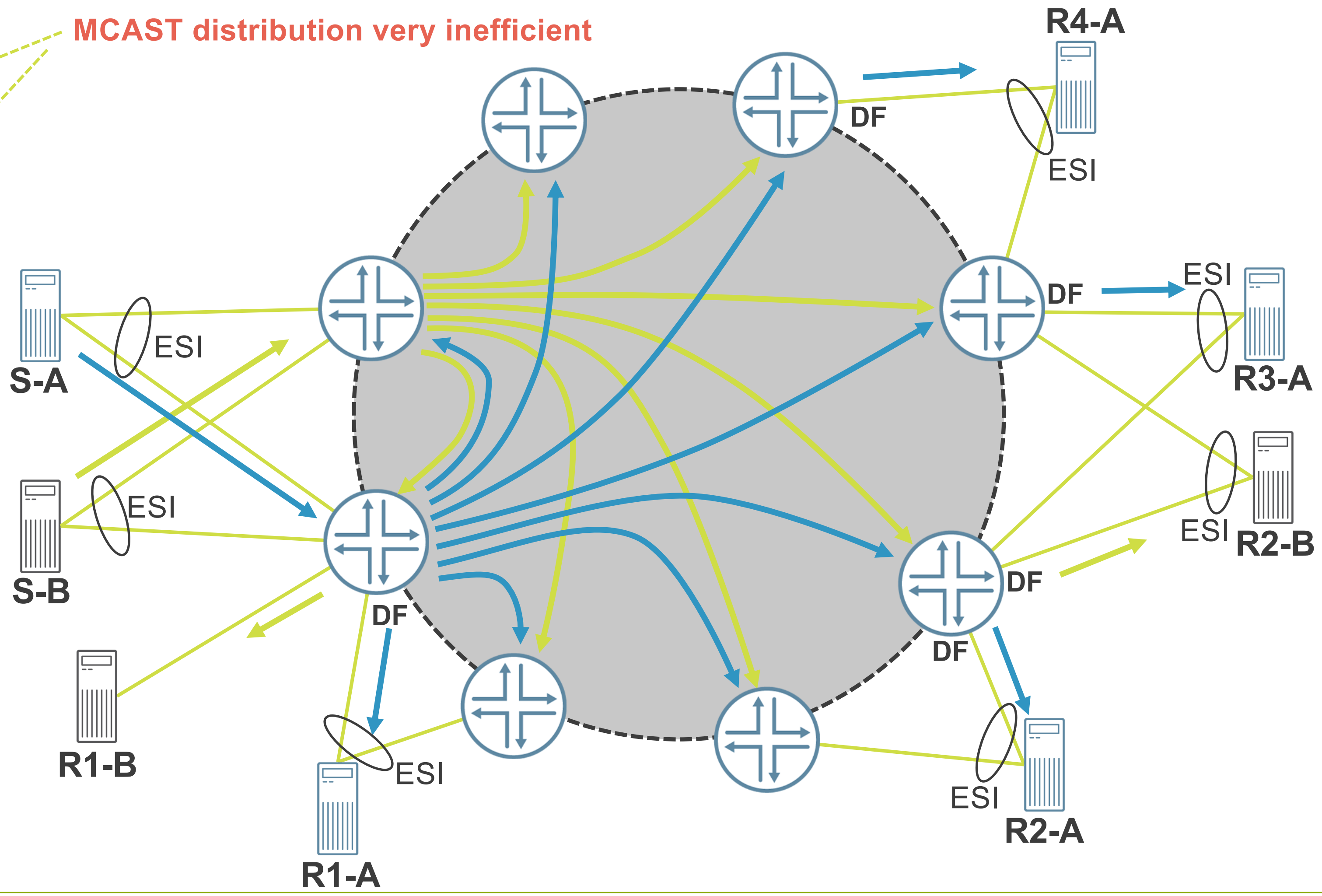
BASIC EVPN MULTICAST DISTRIBUTION (1)

Multicast is delivered from ingress PE to all egress PEs participating in given EVPN via **ingress replication**

Egress PE delivers/blocks MCAST to local receivers based on

- DF/non-DF state
- Local IGMP membership state

MCAST distribution very inefficient



BASIC EVPN MULTICAST DISTRIBUTION (2)

Two aspects of inefficient MCAST distribution in basic EVPN deployments

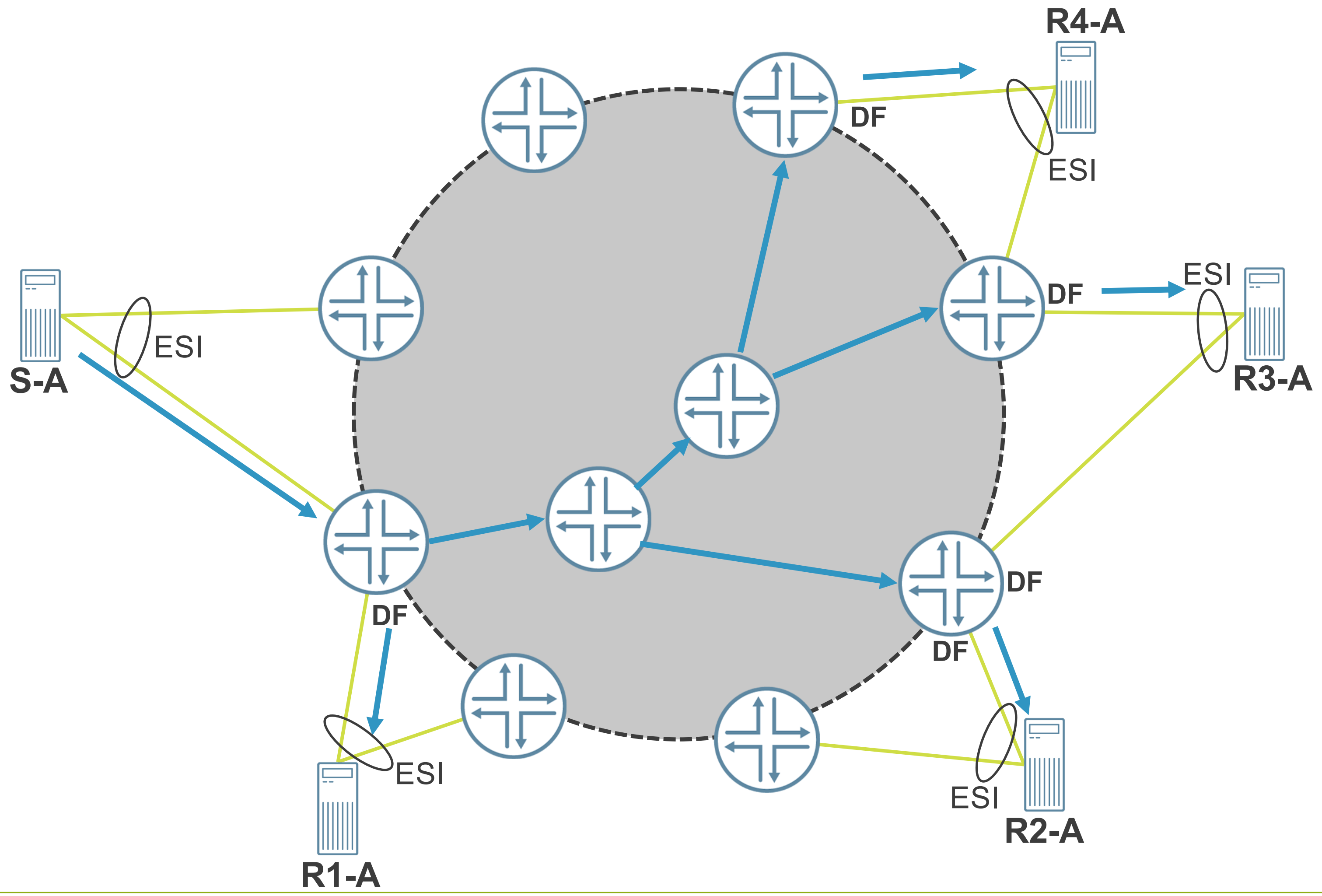
- Ingress replication
  - More efficient replication methods required
    - P2MP (i.e. PIM, mLDP, RSVP, BIER)
    - Assisted Replication

- MCAST distributed to all PEs
  - EVPN creates states based on
    - Data plane or PE-CE control plane (for traffic received from CE)
      - IGMP
    - PE-PE BGP EVPN control plane (for traffic received via EVPN core)
      - BGP EVPN extensions required to accomplish that → SMET (Type 6) Route

EVPN P2MP MULTICAST DISTRIBUTION

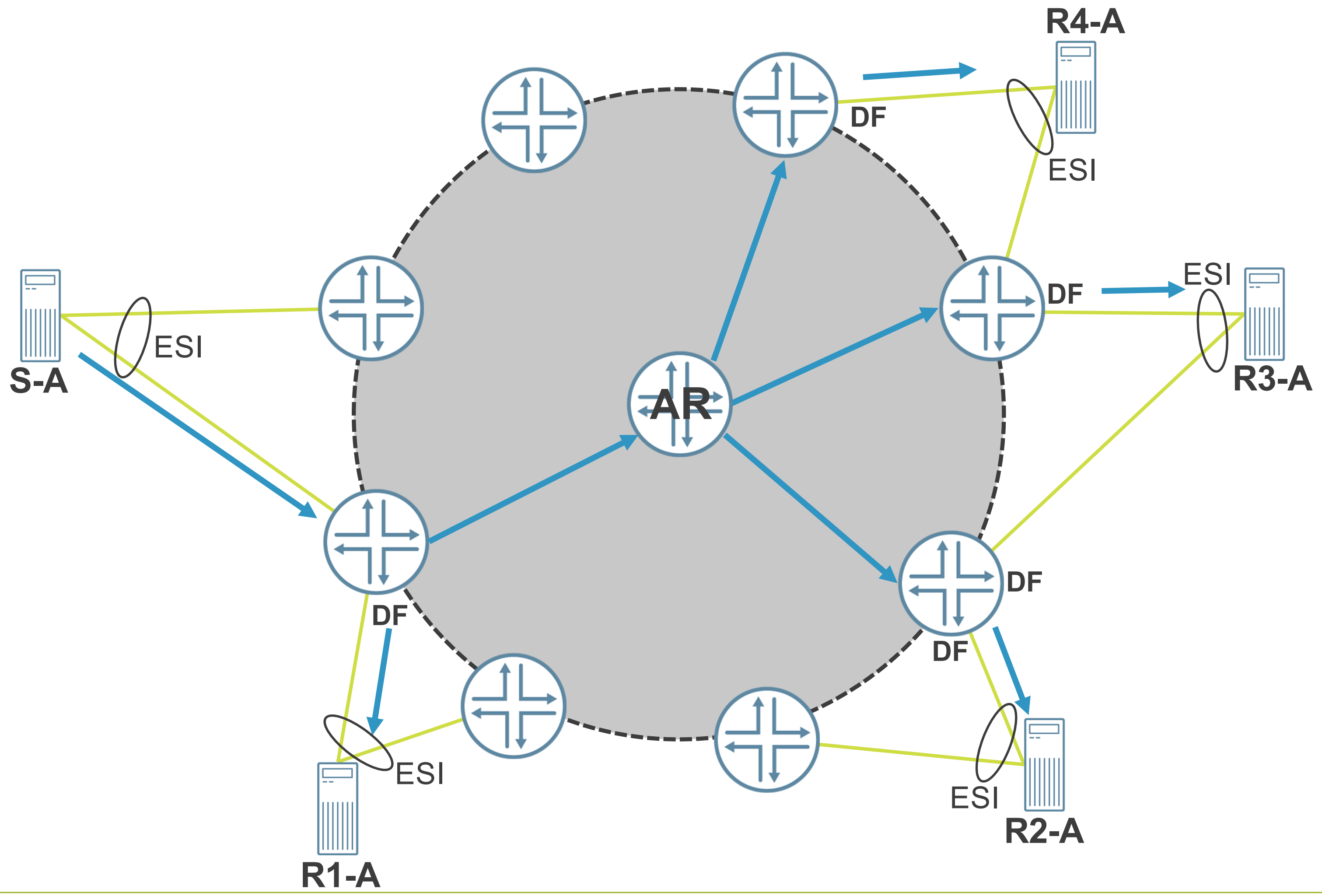
BUM frames are replicated on transit nodes, according to the P2MP structure

- Universally deployable in any arbitrary topology
- Requires consistent P2MP support on all nodes
- Information about P2MP tunnel distributed via Provider Multicast Service Interface (PMSI) attribute in the Inclusive Multicast Ethernet Tag (Type 3) EVPN Route



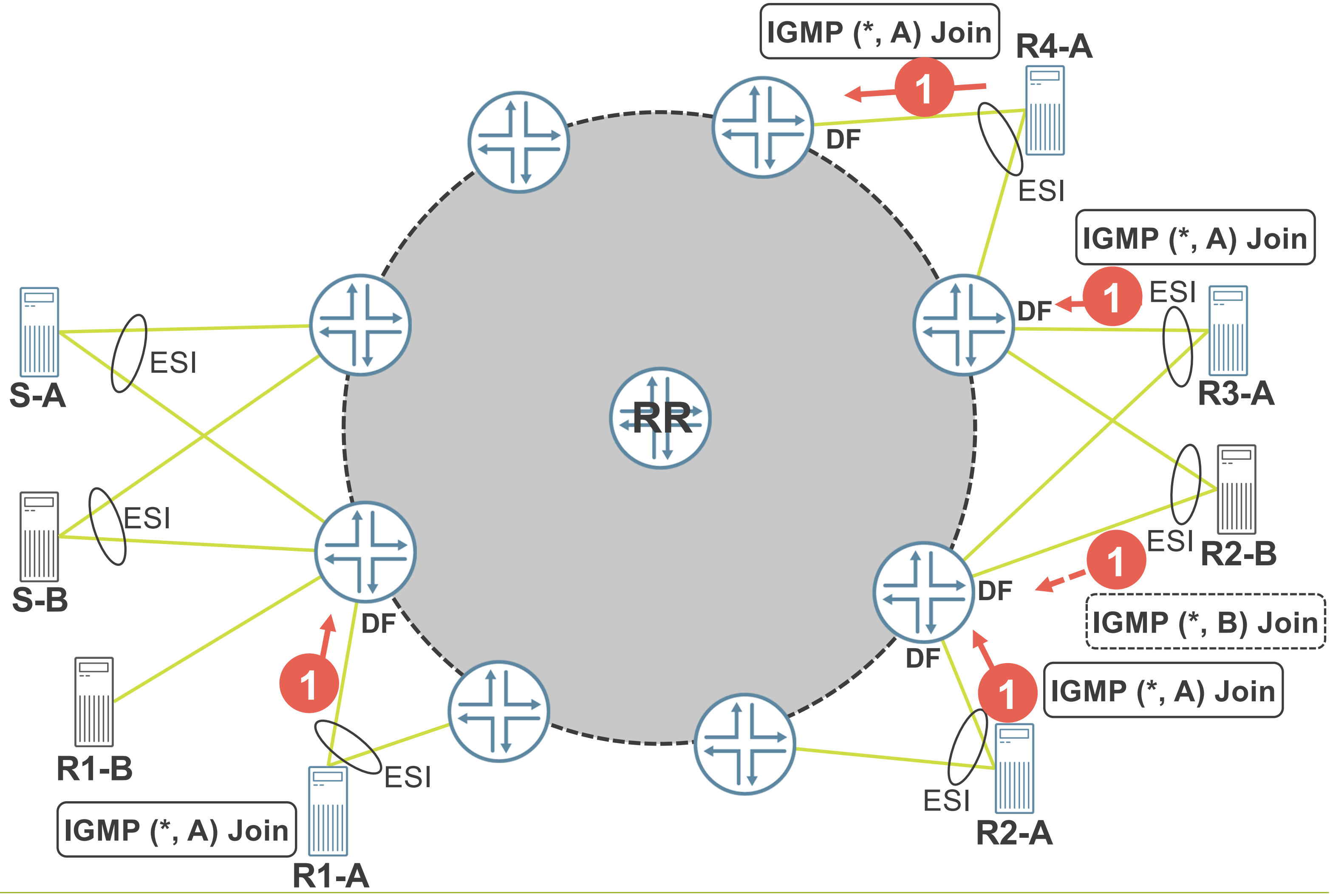
EVPN ASSISTED REPLICATION

- Referred often as "Optimized Ingress Replication"
- Selected (powerful) nodes are designated to perform replication
- Typically suitable to NVO/DC (Leaf/Spine) designs, with powerful Spines, and low performance Leafs



SELECTIVE MULTICAST ETHERNET TAG (SMET) ROUTE (1)

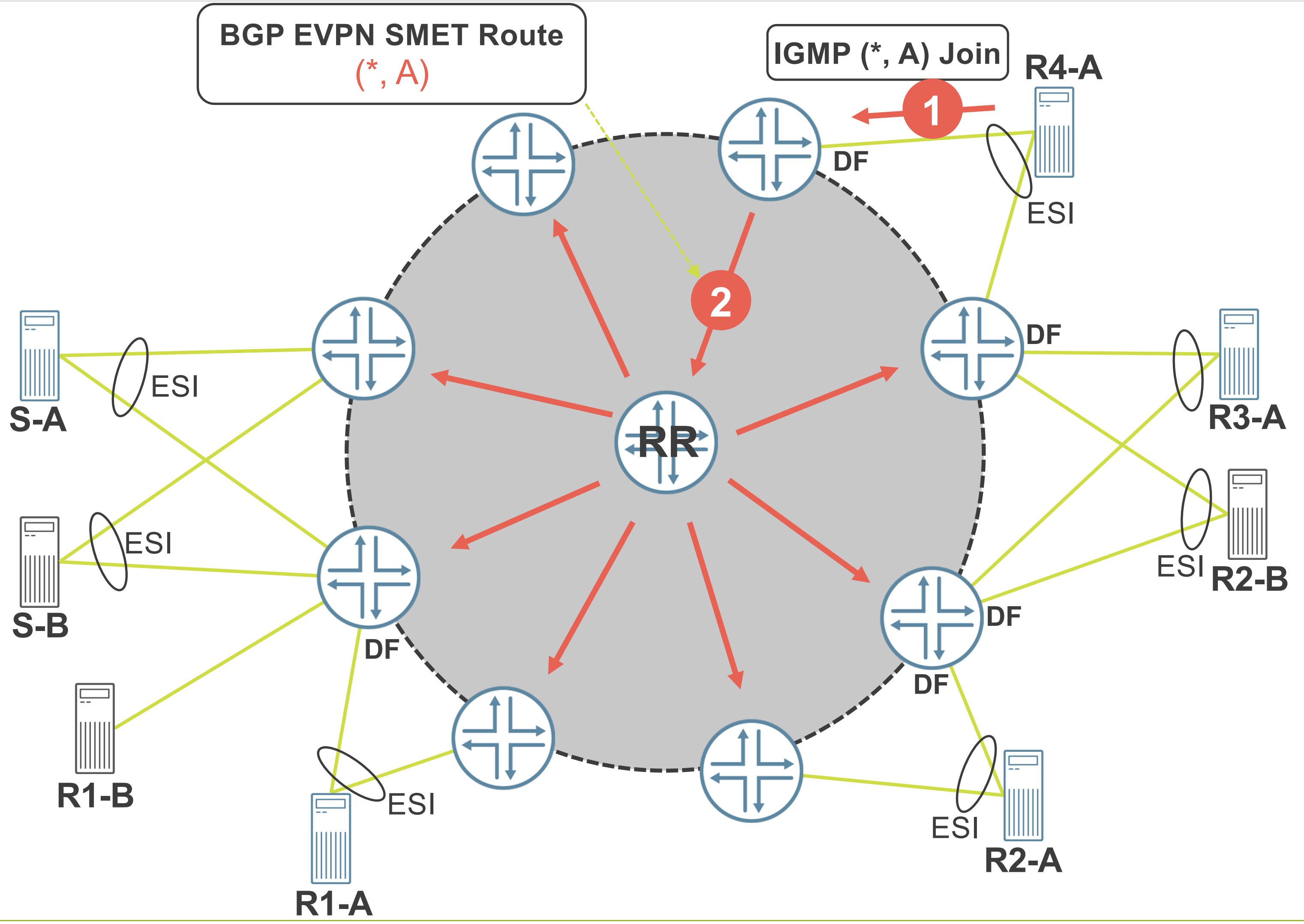
Receives reports the willingness to receive MCAST traffic via standard IGMP (v1/v2/v3) Group Membership ("*Join*") messages



SELECTIVE MULTICAST ETHERNET TAG (SMET) ROUTE (2)

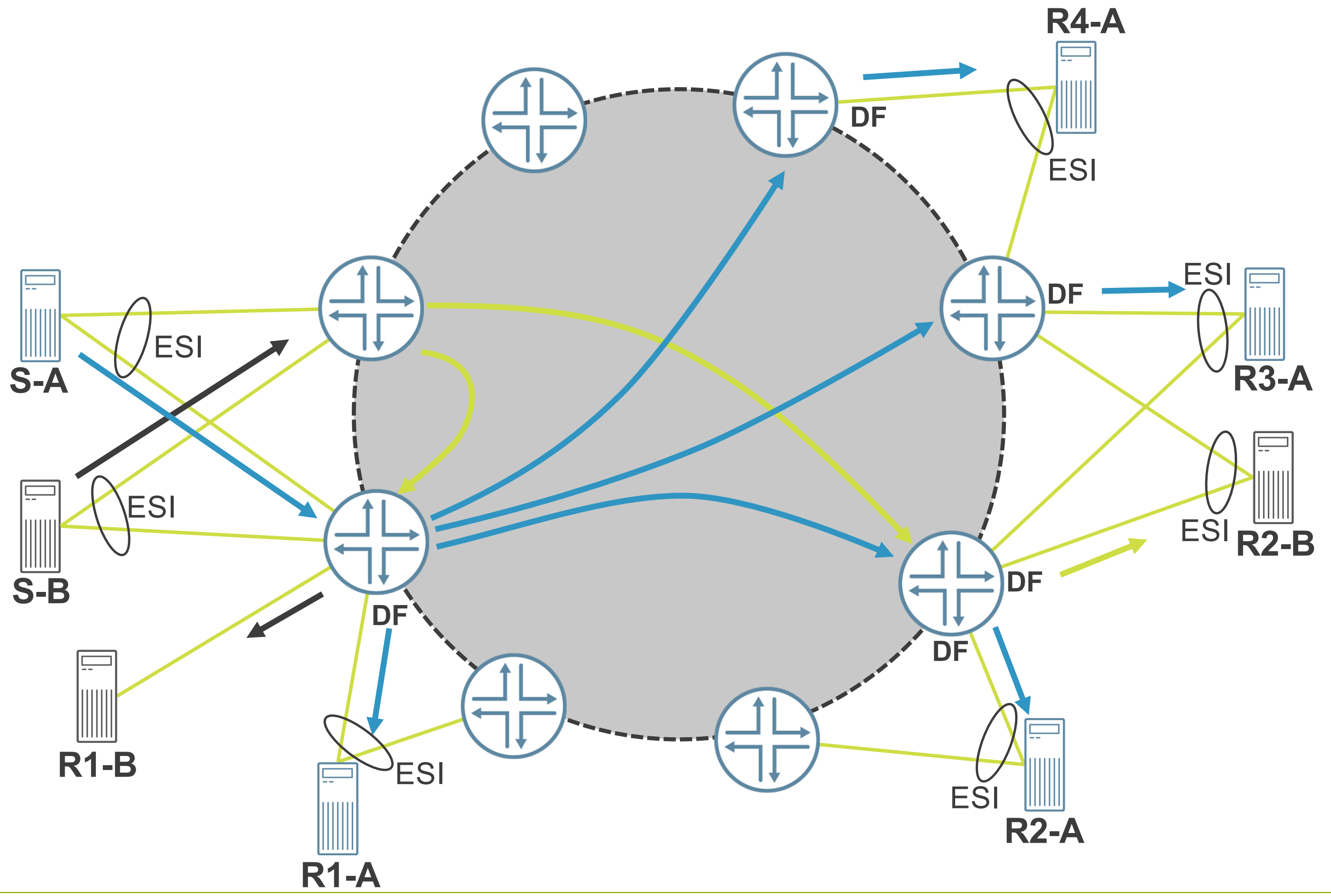
First hop PEs convert IGMP Group Membership messages to BGP EVPN Selective Multicast Ethernet Tag (SMET) messages (Type 6)

- Only R4-A shown, as an example
- Based on that information, all involved PEs are aware, where multicast receivers for specific MCAST flows reside



SELECTIVE MULTICAST ETHERNET TAG (SMET) ROUTE (3)

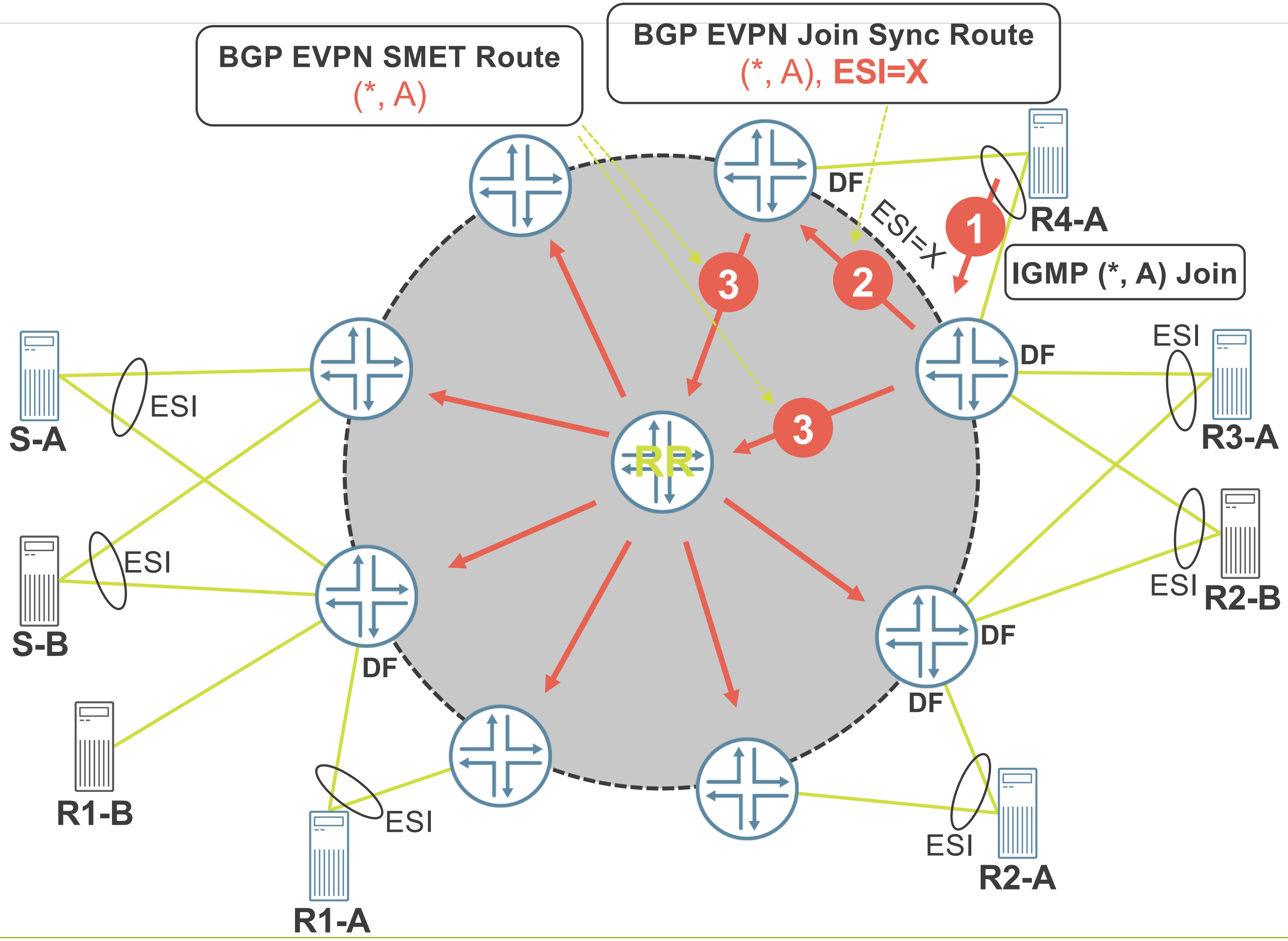
Based on BGP EVPN SMET (Type 6) Route, PEs with attached sources can send MCAST flows to specific PEs only



MULTICAST JOIN SYNC (TYPE 7) ROUTE
MULTICAST LEAVE SYNC (TYPE 8) ROUTE

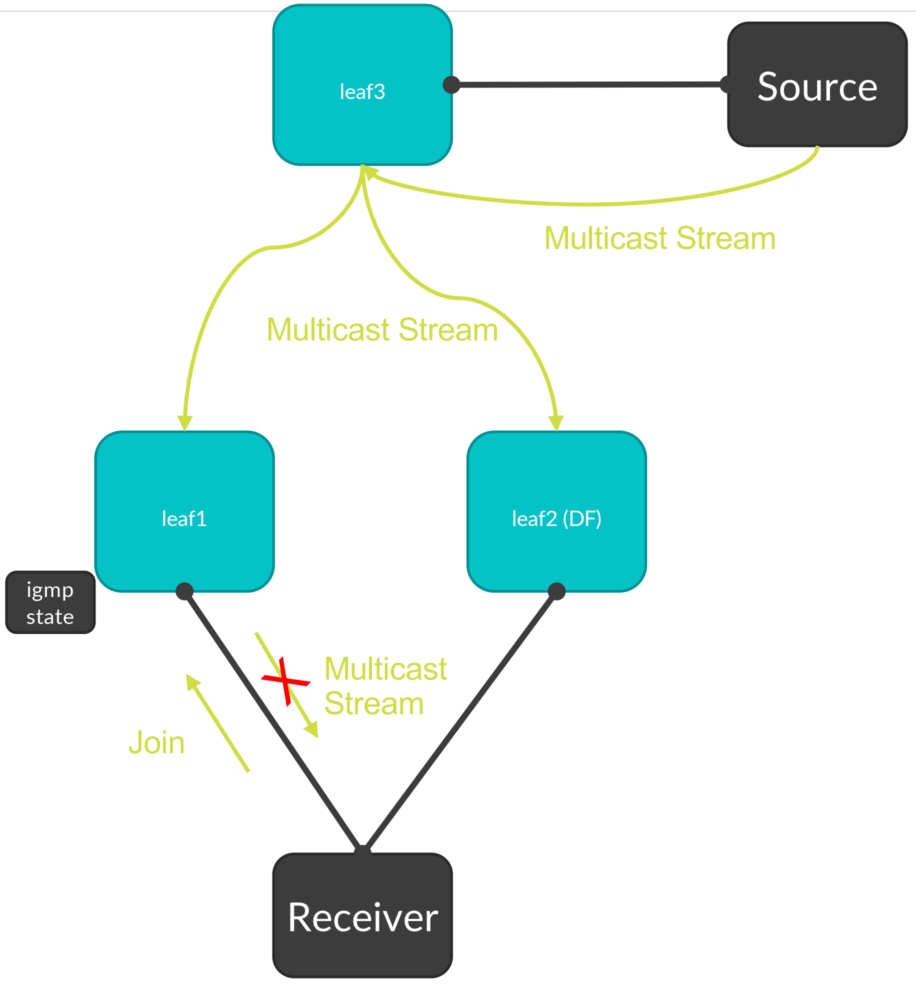
In EVPN A/A multi-homing

1) IGMP Join/Leave might arrive to non-DF
2) It is converted to EVPN Join/Leave Sync (Type 7/8) Route
3) SMET (Type 6) Route announced by DF and nDF based on local IGMP Join or EVPN Join



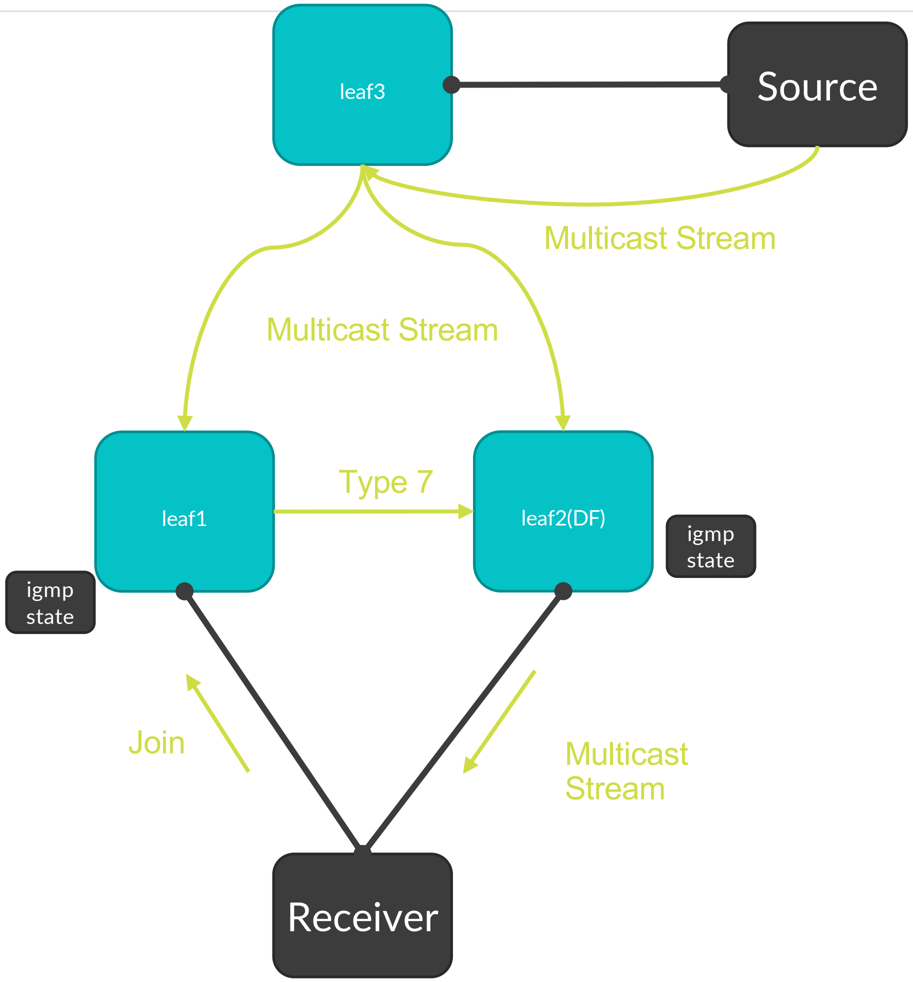
EVPN TYPE-7 JOIN-SYNC ROUTE IN DC ENVIRONMENT

- IGMP join sync message
  - Sync IGMP state between MH Pes
- Without Type 7, If DF does not have IGMP state it could result in multicast traffic starvation



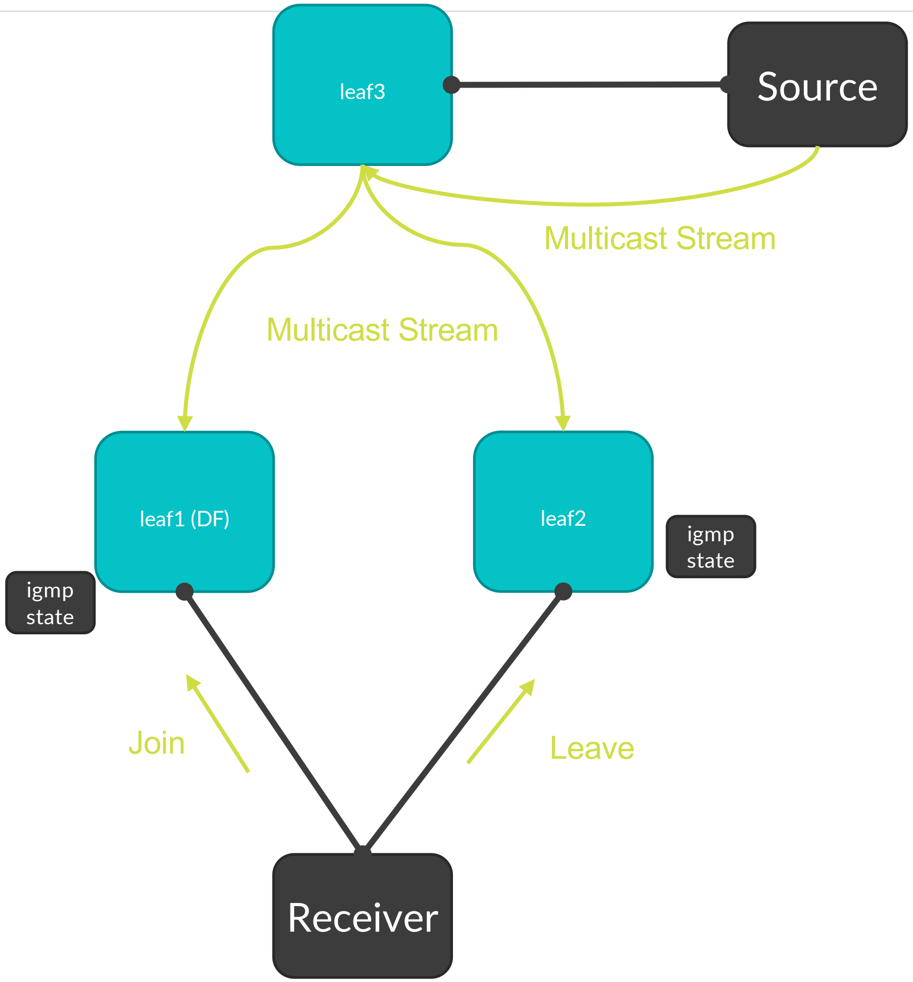
EVPN TYPE-7 JOIN-SYNC ROUTE IN DC ENVIRONMENT

- IGMP join sync message
  - Sync IGMP state between MH Pes
- With Type 7, new DF has the required state to continue forwarding



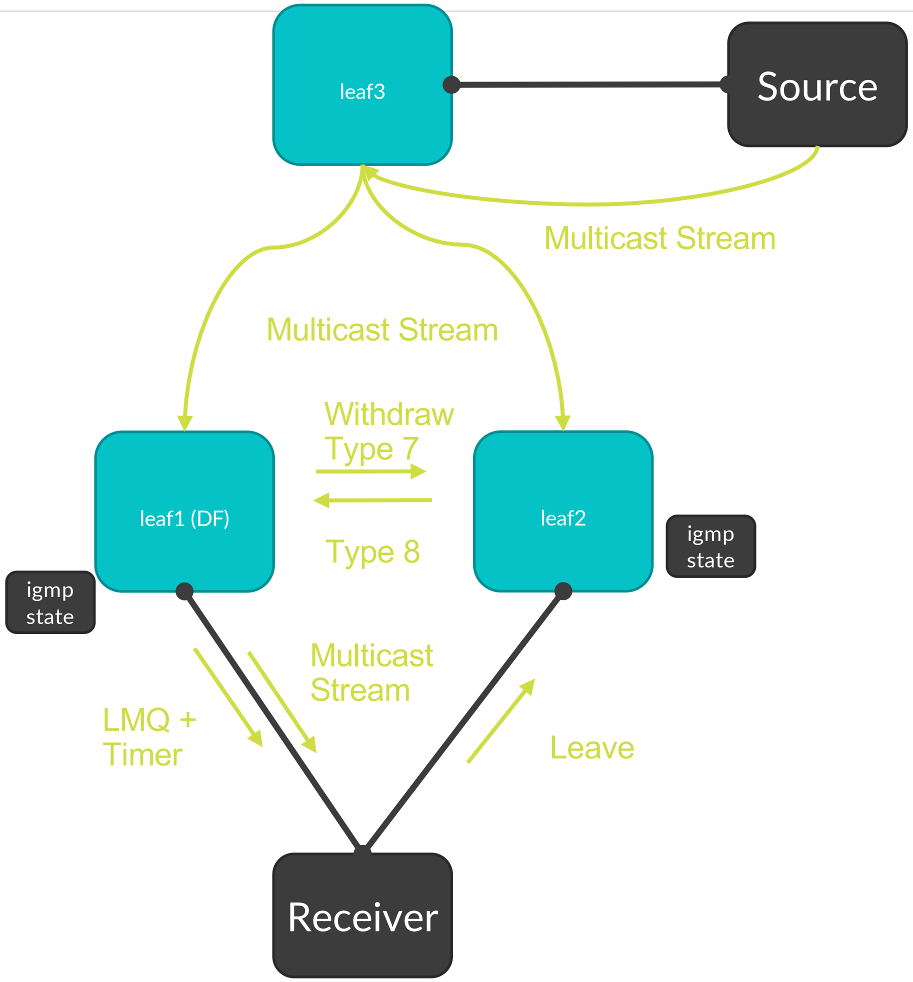
EVPN Type-8 Leave-Sync route in DC environment

- IGMP leave sync message
  - Sync IGMP leave between MH Pes
- Rx sends IGMP leave to different PE than the one it sent join
- PE receiving leave message removes IGMP state however DF is still forwarding



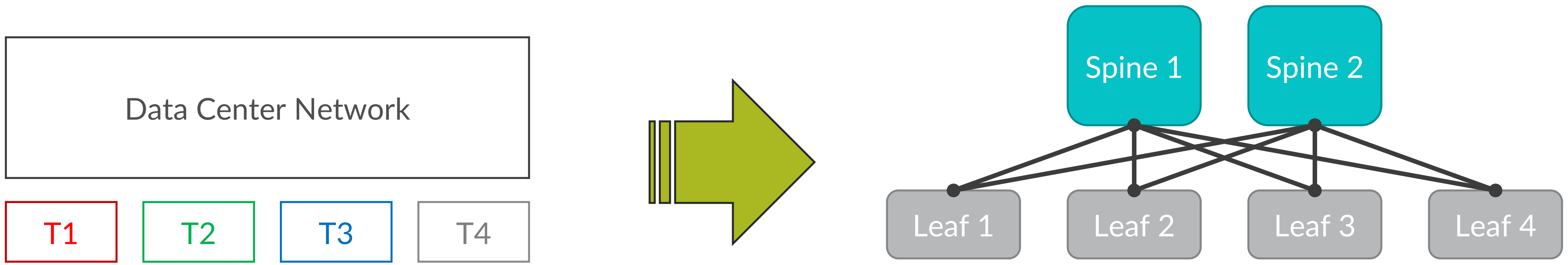
EVPN Type-8 Leave-Sync route in DC environment

- MH PE receiving IGMP leave sends Type 8 to sync the leave message
- Other PE sends Leave Membership Query and starts timer
- Withdraw Type 7 if no response received and timer expired



DC architectures using EVPN

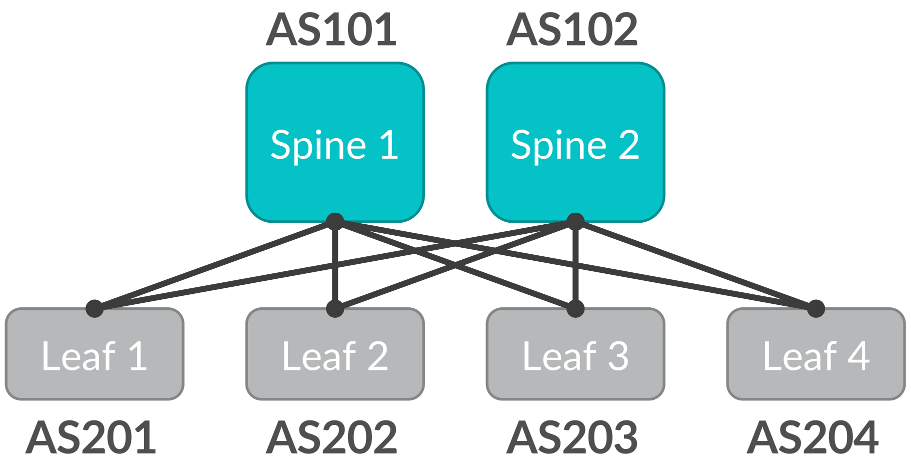
MULTI-TENANT DATACENTER

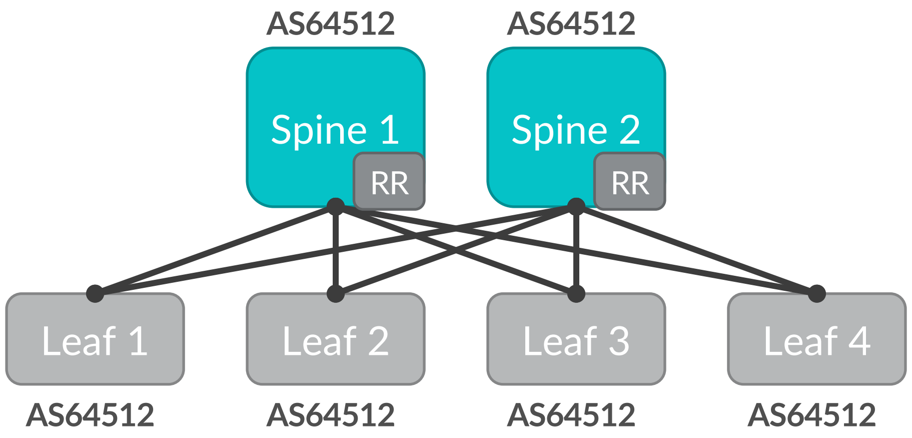


Multi-Tenant Data Center

- CLOS / IP Fabric 3-stages
- Provide isolation between tenants
- Multiple subnets per tenant
- Provide L2 and L3 transit
- Physical and virtual workloads

VXLAN FABRIC BGP RECOMMENDATION





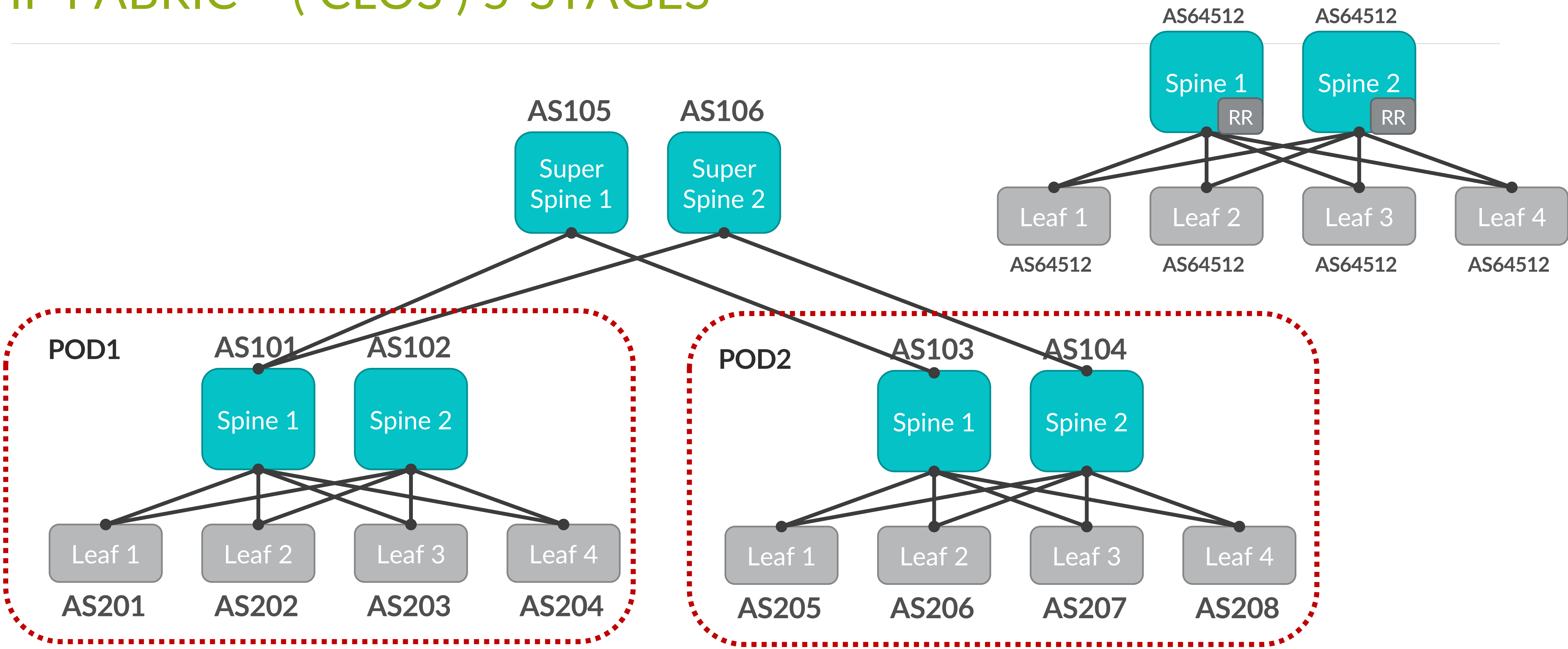
EBGP for Substrate / Underlay

- Simple design
- eBGP bound to physical interfaces
- BGP ASN per switch
- Export loopback prefixes for EVPN
- No IGP required
- Topology-aware EBGP
- IP Fabric

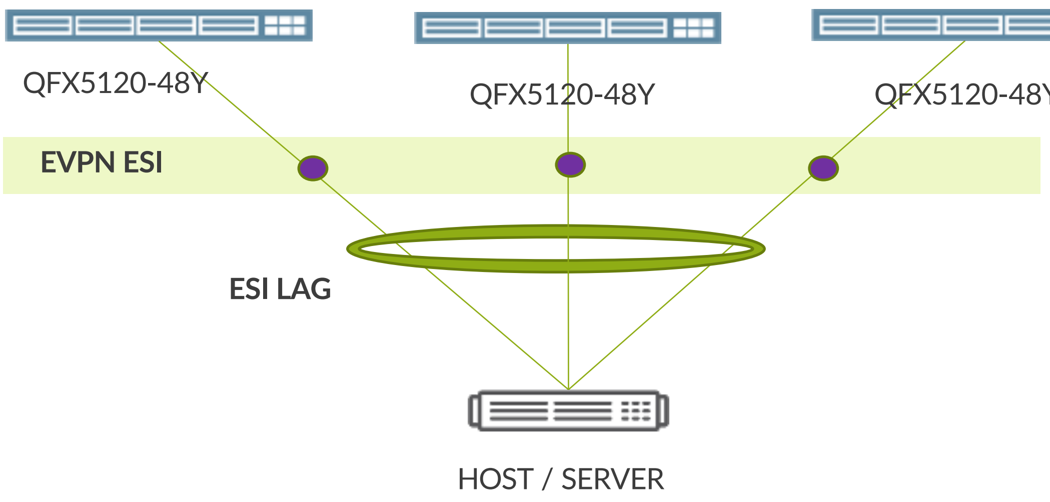
IBGP for EVPN / Overlay

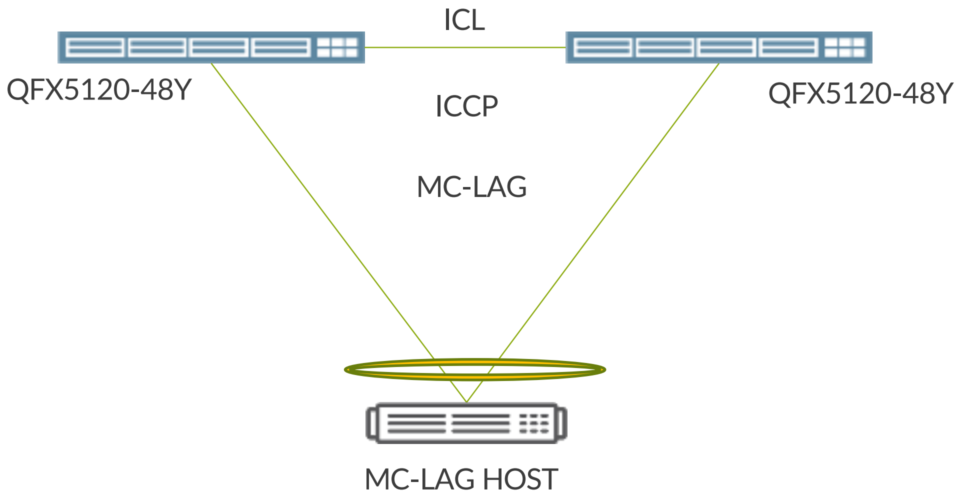
- Simple design
- iBGP bound to loopbacks
- Single BGP ASN
- RR avoids full-mesh peering
- MAC learning + ESI
- Topology independent

IP FABRIC – (CLOS) 5-STAGES



HOST CONNECTIVITY OPTIONS





ESI LAG	MC-LAG
Active-Active multi-homing	Active-Active dual homing
All interfaces part of same ESI correspond to same L2 domain. Type 7, Type 8 support for multicast, efficient BUM traffic handling	Redundancy and load balancing between two MC-LAG peers Loop-free L2 network without STP

ESI CONFIG

```
ae0 {
      flexible-vlan-tagging;
      mtu 9192;
      encapsulation extended-vlan-bridge;
      esi {
          00:01:01:01:01:01:01:01:01:01;
          all-active;
      }
      aggregated-ether-options {
          link-speed 10g;
          lacp {
              active;
              system-id 00:00:00:00:00:01;
          }
      }
      unit 200 {
          vlan-id 200;
      }
  }
```

```
vlans {
    MULTI-HOMED-VNI {
        interface ae0.200;
        vxlan {
            vni 200;
            ingress-node-replication;
        }
    }
```

IDENTIFY ESI AND TYPE-2

```
bgp.evpn.0: 72 destinations, 119 routes (72 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

1:3.3.3.3:0::010101010101010101::FFFF:FFFF/192 AD/ESI
                   *[BGP/170] 02:09:12, localpref 100, from 4.4.4.4
                      AS path: I, validation-state: unverified
                       to 172.25.1.2 via et-0/0/48.0
                    >  to 172.26.1.2 via et-0/0/49.0
                    [BGP/170] 02:09:04, localpref 100, from 5.5.5.5
                      AS path: I, validation-state: unverified
                       to 172.25.1.2 via et-0/0/48.0
                    >  to 172.26.1.2 via et-0/0/49.0
1:3.3.3.3:1::010101010101010101::0/192 AD/EVI
                   *[BGP/170] 02:09:13, localpref 100, from 4.4.4.4
                      AS path: I, validation-state: unverified
                       to 172.25.1.2 via et-0/0/48.0
                    >  to 172.26.1.2 via et-0/0/49.0
                    [BGP/170] 02:09:04, localpref 100, from 5.5.5.5
                      AS path: I, validation-state: unverified
                       to 172.25.1.2 via et-0/0/48.0
                    >  to 172.26.1.2 via et-0/0/49.0
```

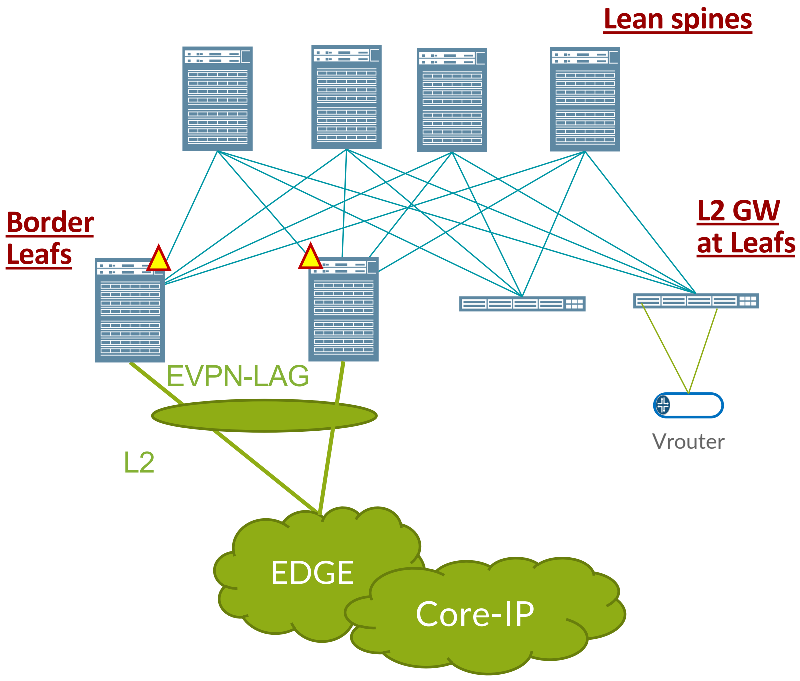
IDENTIFY ESI AND TYPE-2

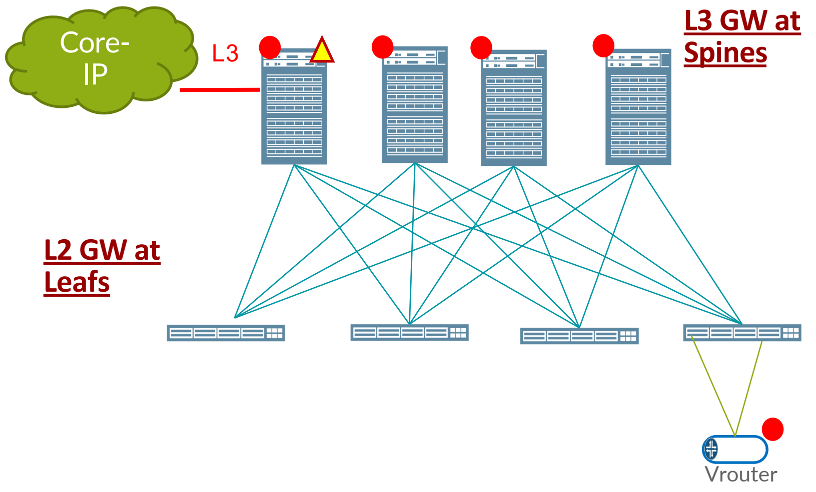
```
2:3.3.3.3:1::200::00:11:01:00:00:07/304 MAC/IP
                           *[BGP/170] 03:57:09, localpref 100, from 4.4.4.4
                              AS path: I, validation-state: unverified
                               to 172.25.1.2 via et-0/0/48.0
                            >  to 172.26.1.2 via et-0/0/49.0
                            [BGP/170] 03:57:09, localpref 100, from 5.5.5.5
                              AS path: I, validation-state: unverified
                               to 172.25.1.2 via et-0/0/48.0
                            >  to 172.26.1.2 via et-0/0/49.0


2:3.3.3.3:1::200::00:11:01:00:00:01::10.1.1.2/304 MAC/IP
```

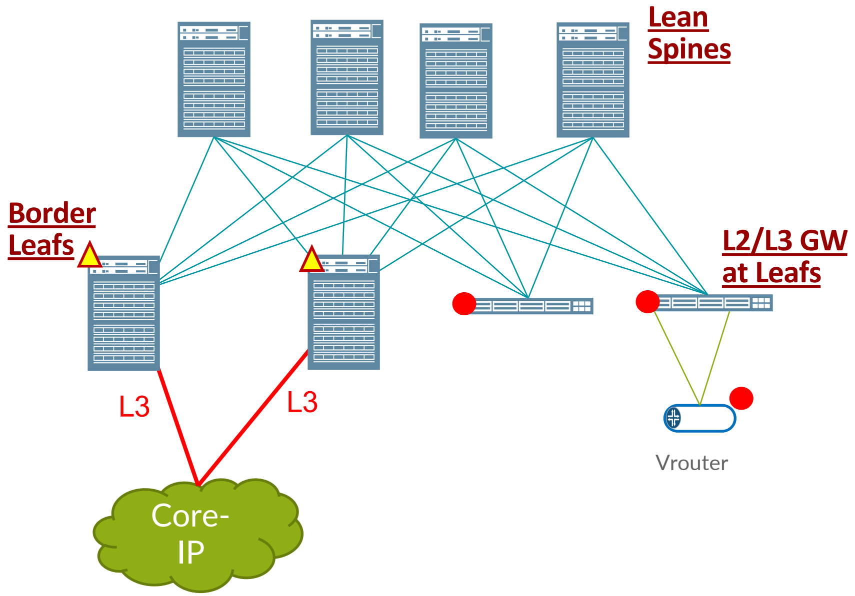
MAC/IP for Proxy ARP on leaf switch

EVPN-VXLAN REFERENCE ARCHITECTURES IN THE DATA CENTER

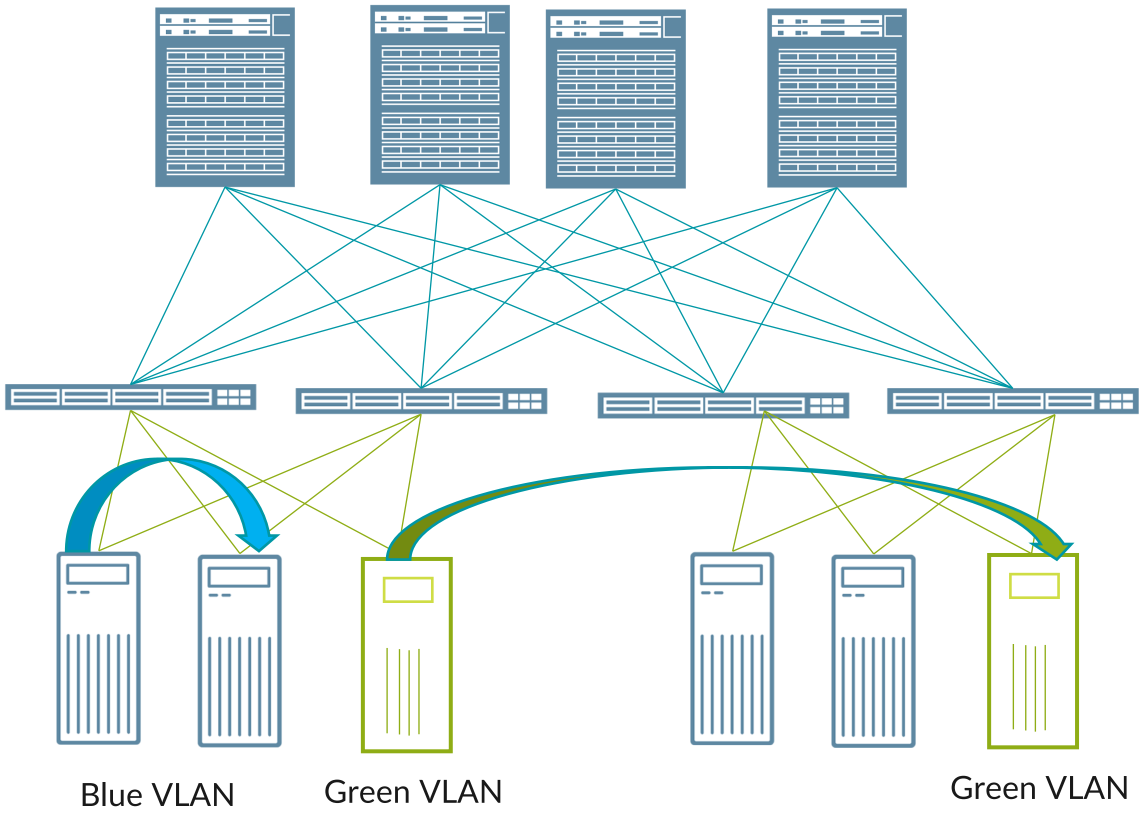
Bridged Overlay - BO	Centrally Routed Bridging - Centrally Routed	Edge Routed Bridging - Edge Routed







BRIDGED OVERLAY – BO



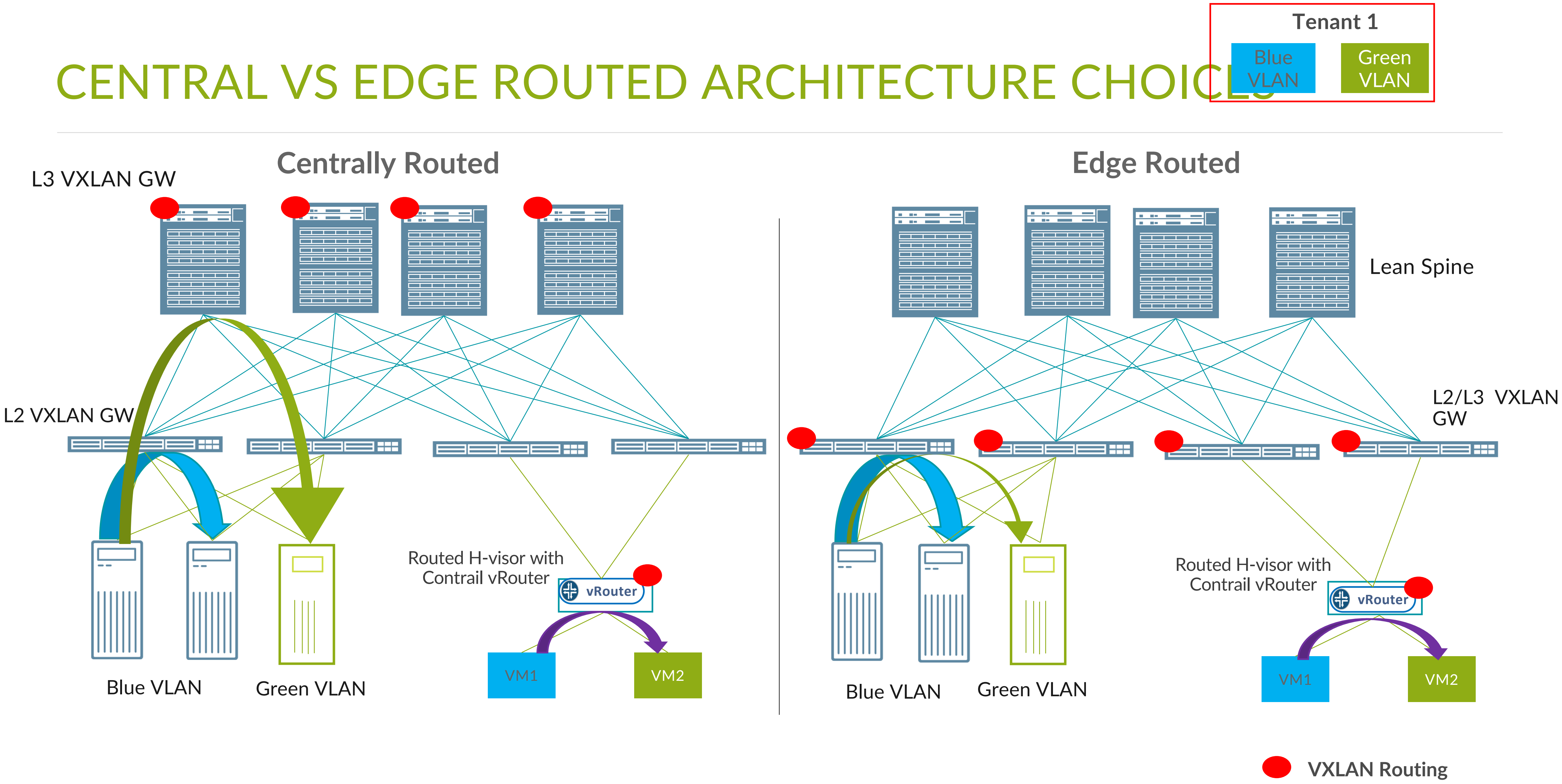
What is BO?

- A bridged overlay provides
  Ethernet bridging between leaf devices in an EVPN network
- This overlay type simply extends VLANs between the leaf devices across VXLAN tunnels

Why BO?

- No IP gateway migration/ IP gateway are managed by the external tenant

CENTRAL VS EDGE ROUTED ARCHITECTURE CHOICES



OBRIGADO!

Eduardo Haro, Juniper System Engineer
eharo@juniper.net

Parceria

JUNIPER NETWORKS | Engineering Simplicity

Realização

ceptro.br nic.br