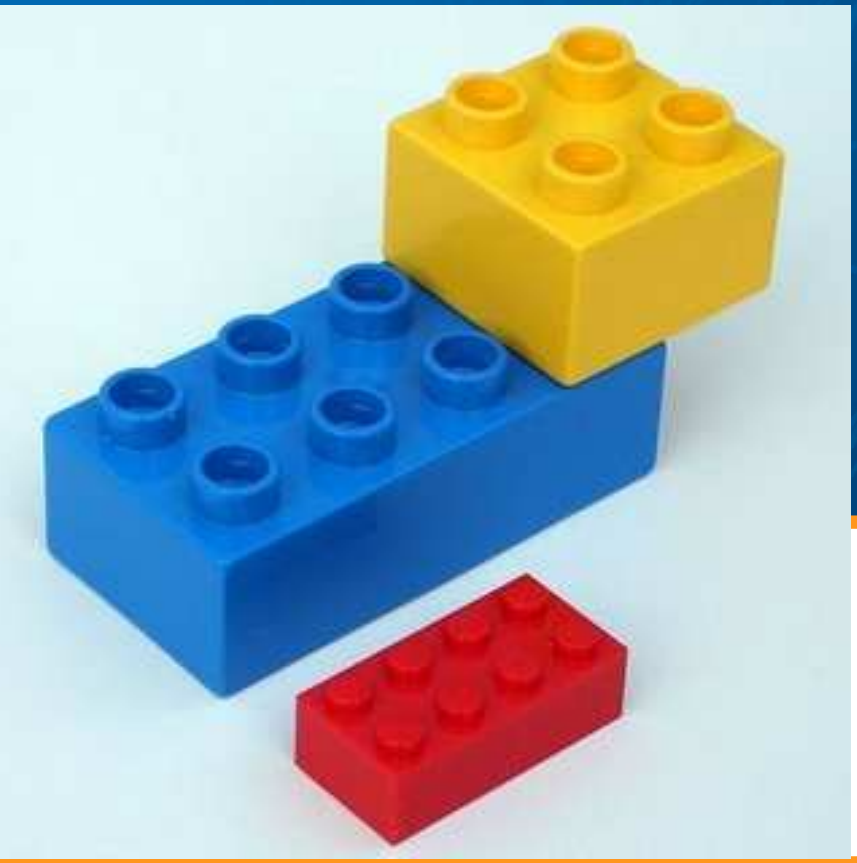


SUBSECOND END TO END SERVICE RESTORATION

Emil Gągała
Sr. Systems Engineer

PLNOG, Kraków, 10.09.2009

Credits

Most of intellectual work with features covered in this session was done by Hannes Gredler from JUNOS Software Engineering.

Thanks Hannes.

... and thanks to Robert Raszuk for being technical advocate of these features implementation

This session …

- ***is about a novel MPLS interregion local-repair, restoration approach that protects both transport and service LSPs with an upper boundary of 50ms, independent of network scale.***

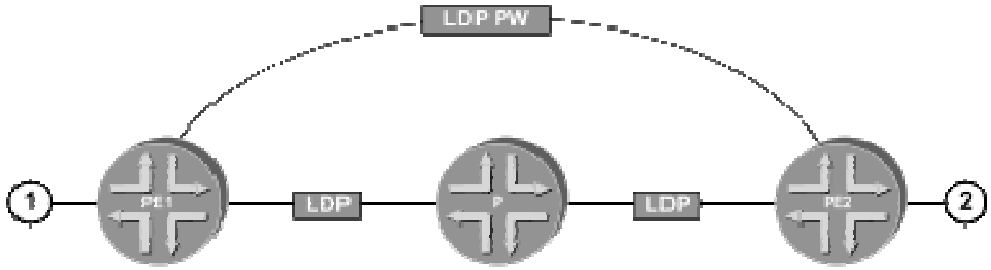
Agenda

- What's the problem, at all ?
- Loop-free Alternate
  - Coverage extension
- Protecting Tunnel Endpoints
  - Generalized Interdomain model
  - Data plane
  - Example LDP, RSVP, 2547 VPN
- FIB Aggregation

What is the problem, at all ?

Why 50ms service protection ?

- Service is just itself a transport for the application
- Application may or may not require 50ms resilience
- Service may be transport for others.
- Transport inferred *magic* boundary

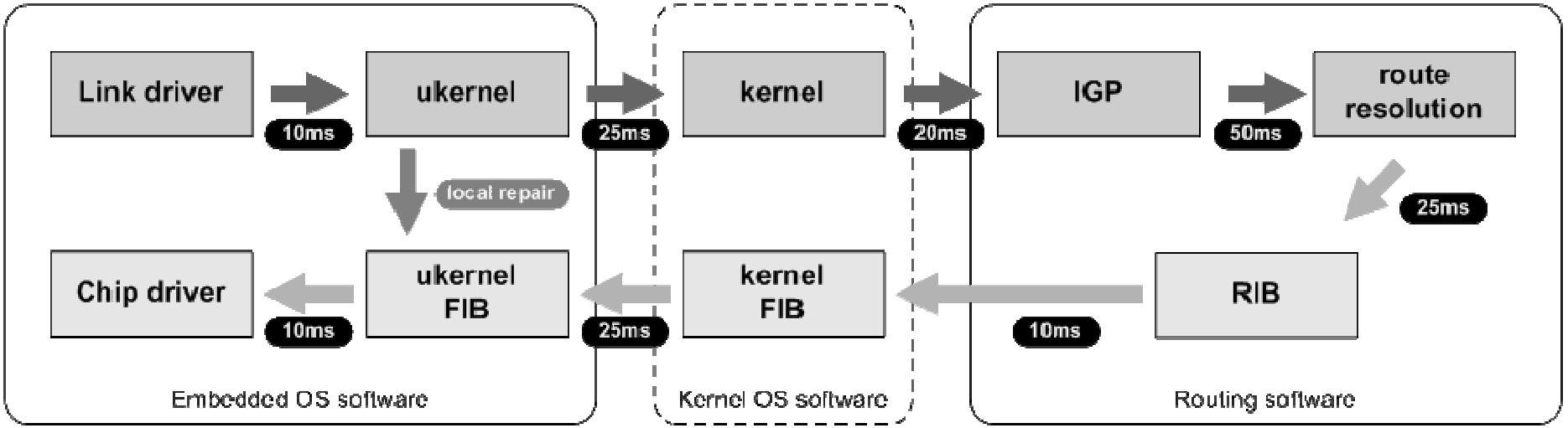


Christmas wish list for service convergence

- End to end service restoration (<50ms)
  - Control-plane based solutions ruled out
  - Multi-hop BFD solutions ruled out
  - Local repair only choice left
- Local repair cost O(1)
  - no dependency on # FIB entries
- Node protection
- Infinite scale (>50K of service endpoints)
- Protect Tunnel Ends (here it gets tough !)
  - Interdomain transport
  - Service (VPN, VPLS, L2VPN)
  - Requires IP local repair

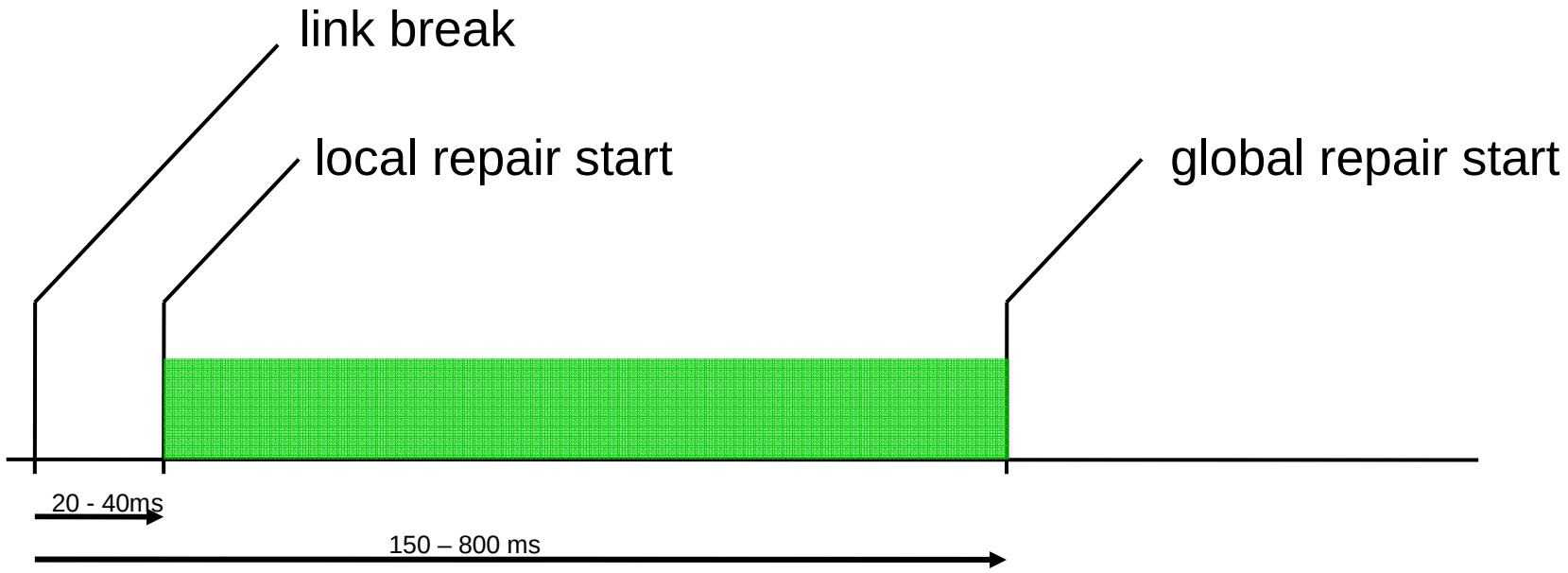
Router event propagation

- **Example: Link down event**



Timeline

- **Local-repair complements global repair**
- **Start times depending on system load and platform (single-PFE, multi-PFE, CPU)**



Network scaling drivers

- FRR available for RSVP since JUNOS 3.4
- Requires PE to PE full mesh for transport plane protection
  - LDP solves this using sink-trees
- RSVP N^2 Problem
  - 200 PEs = app. 39800 LSPs + detours & bypasses + make-before-break paths + secondary standby LSP...
  - Fixable using LSP hierarchy
- This address only transport area, but not service
  - Fail of ingress or egress not covered

LOOP–FREE ALTERNATES

Convergence in IGP needs to involve Control Plane

- Link-fails
- Signal this event to neighbours via IGP
- Recompute new next-hops for all affected prefixes
  - Install result into forwarding-plane
  - Forwarding-plane can use newly installed next-hop
- LFA goal
  - Speedup reconvergence heavily by enabling the forwarding-plane to select new next-hop without interaction with controlplane

What does LFA provide

- LFA brings Fast-reroute capabilities to LDP/IGP
- Does allow <50ms failovertimes
  - Is not dependent from neighbour-support
- Another site-aspect is, that LFA is fully NSR-capable
  - LFA w/LDP is giving us NSR-support for l2vpn, VPLS and L3VPN
  - P2MP are unbeatable when it comes to multicast-transport for either VPLS or L3VPN (ngMVPN). P2MP-LSP's do support FRR facility-backup with link-protection.
  - So for multicast in MPLS-VPN's RSVP-P2MP is still the recommended solution

SPF route calculation revisited

SPF route calculation steps

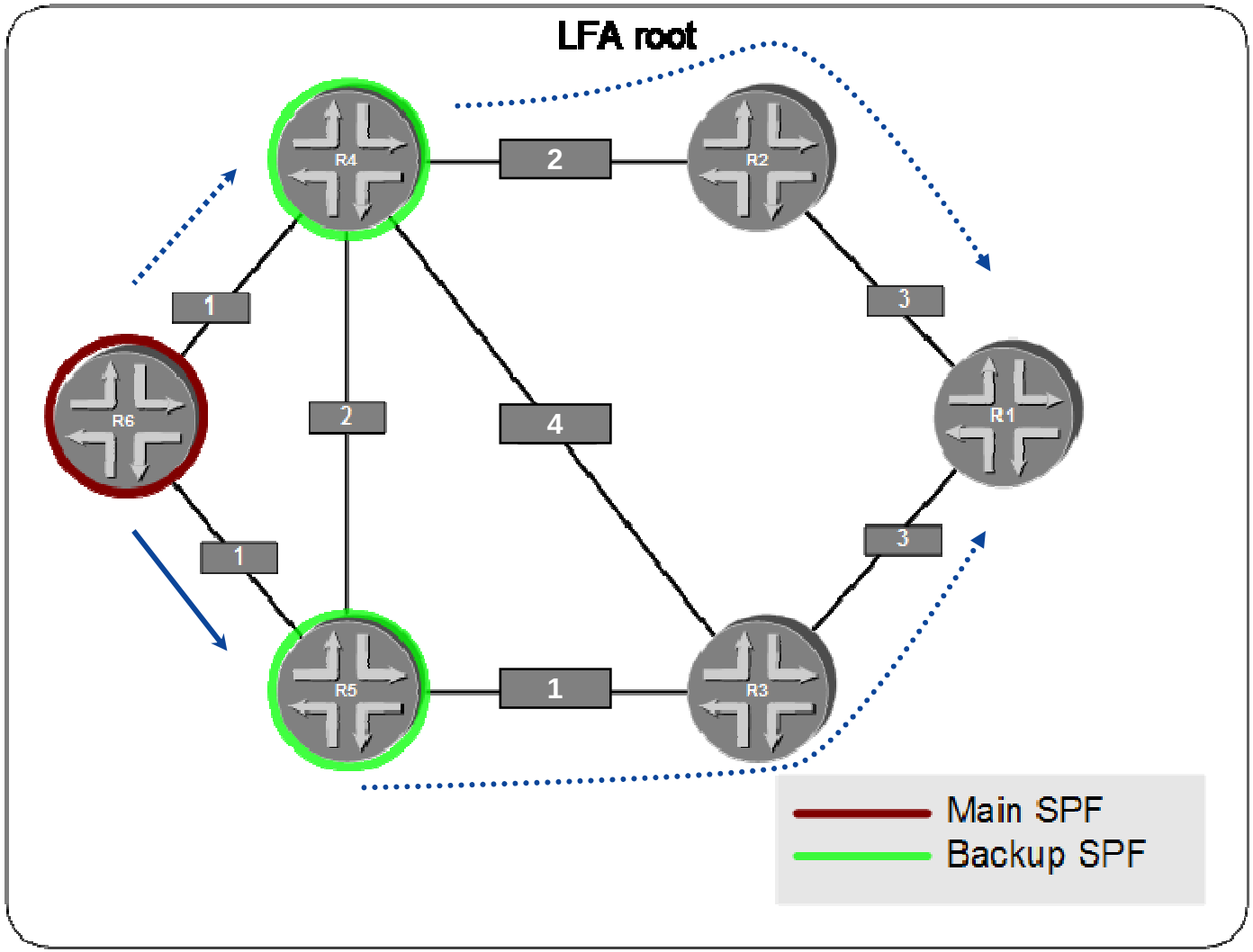
- **SPF init**
  - Reset all nodal cost to infinity
  - Reset our nodal cost to zero
- **Dijkstra's Algorithm**
  - Explore all edges (= neighbors)
  - Put a new or better path on a candidate list
  - Extract the minimum cost node from the *candidate* (=tentative) list and put it to the *result* (= path) list
- **RIB Maintenance**
  - Update all prefixes and pick the best prefix
  - Inherit nexthops from best prefix originator
- **FIB installation**
  - Incremental update for routes with changed nexthops

Loop free alternates basics

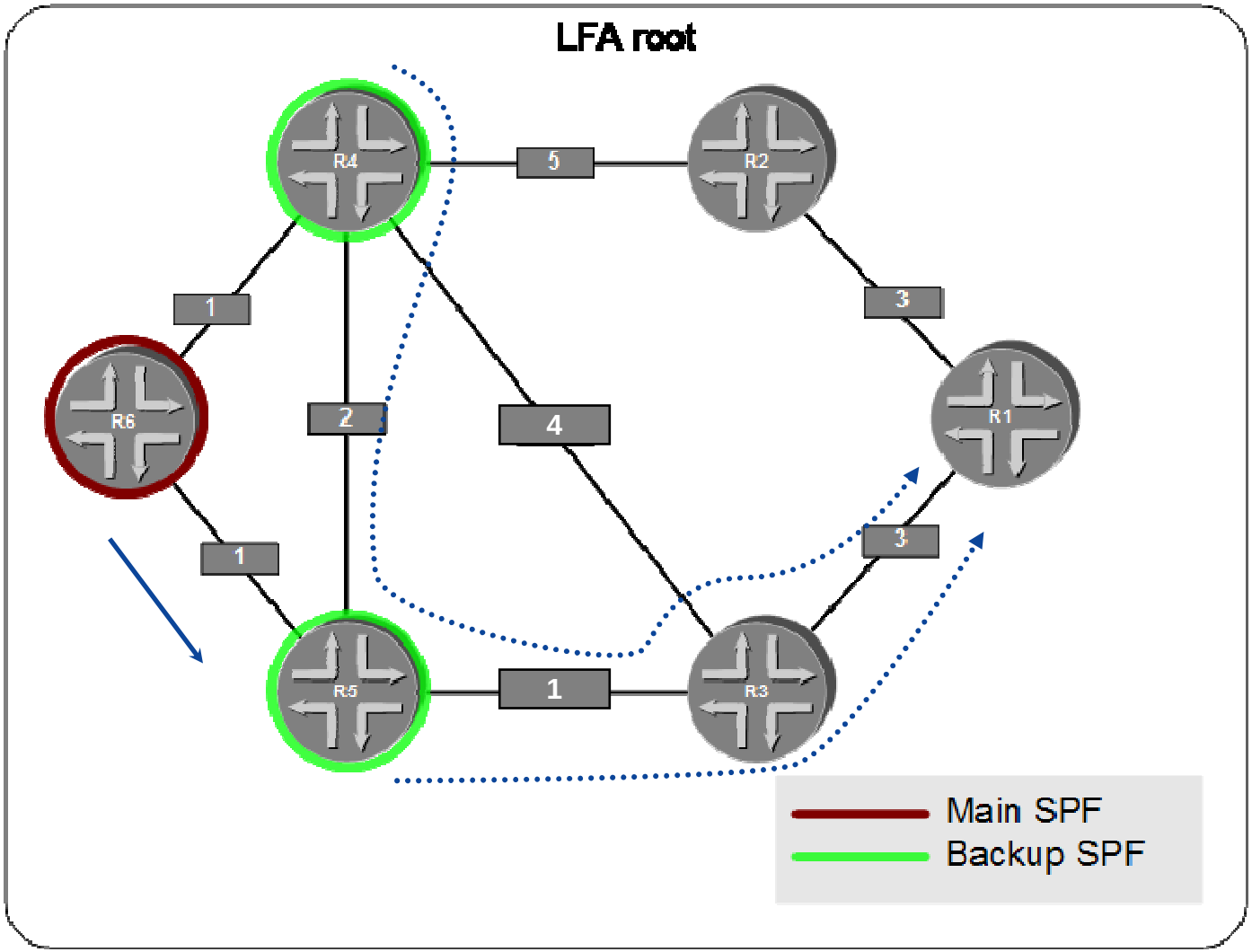
Loop-free Alternates (LFA)

- **Adds fast-reroute (FRR) capability to IS-IS, OSPF and LDP**
  - **Normally only best nodal path is used for RIB walks**
- **Add a non-best (albeit loop free) path for backup purposes.**
- **How ?**
  - **Shared, common link state database (!)**
  - **Place the SPF root at your neighbors**

SPF Roots & LFA illustrated



SPF Roots & LFA illustrated



Configuration

Configuration

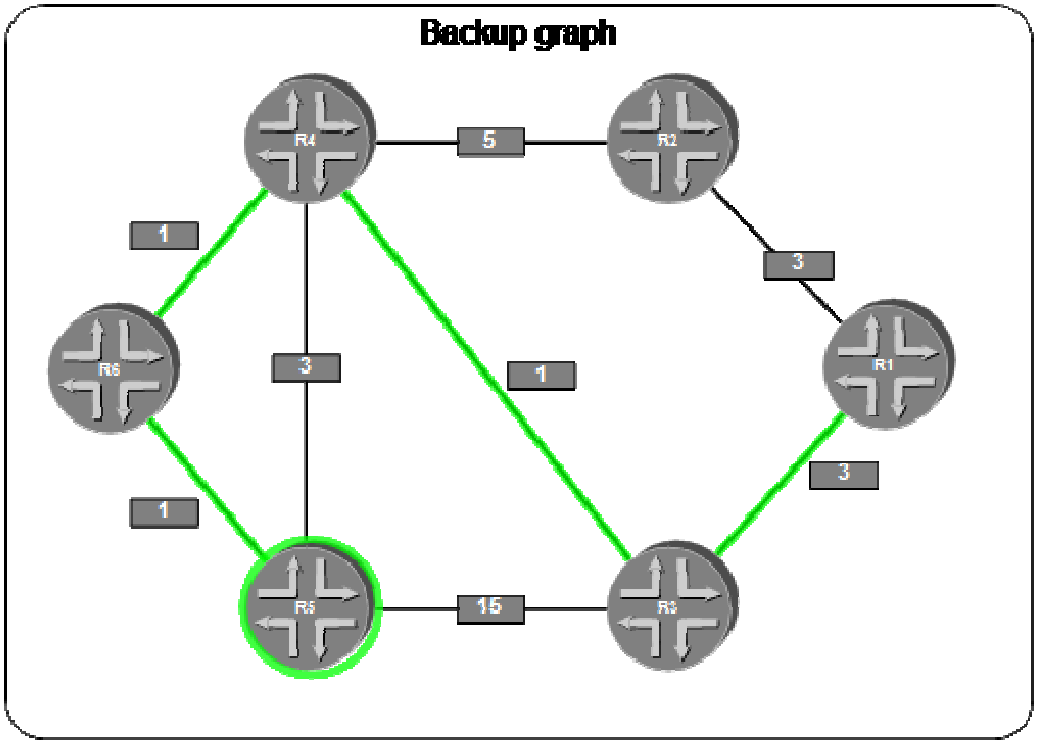
```
protocols {
    isis {
        interface so-1/0/0 {
            nodelink-protection;
        }
        interface so-1/0/0 {
            link-protection;
        }
        interface so-7/0/0 {
            no-eligible-backup;
        }

    }
}
```

Implementation

The Tracklist

- List of *important nodes* passed by
- Build for each neighbor (potential backup next-hop).
- Helps to pick good backup next-hops

- Important nodes
  - Ourselves
  - 1-Hop Neighbors (for node-link-protection)
  - Tail end of MPLSPs

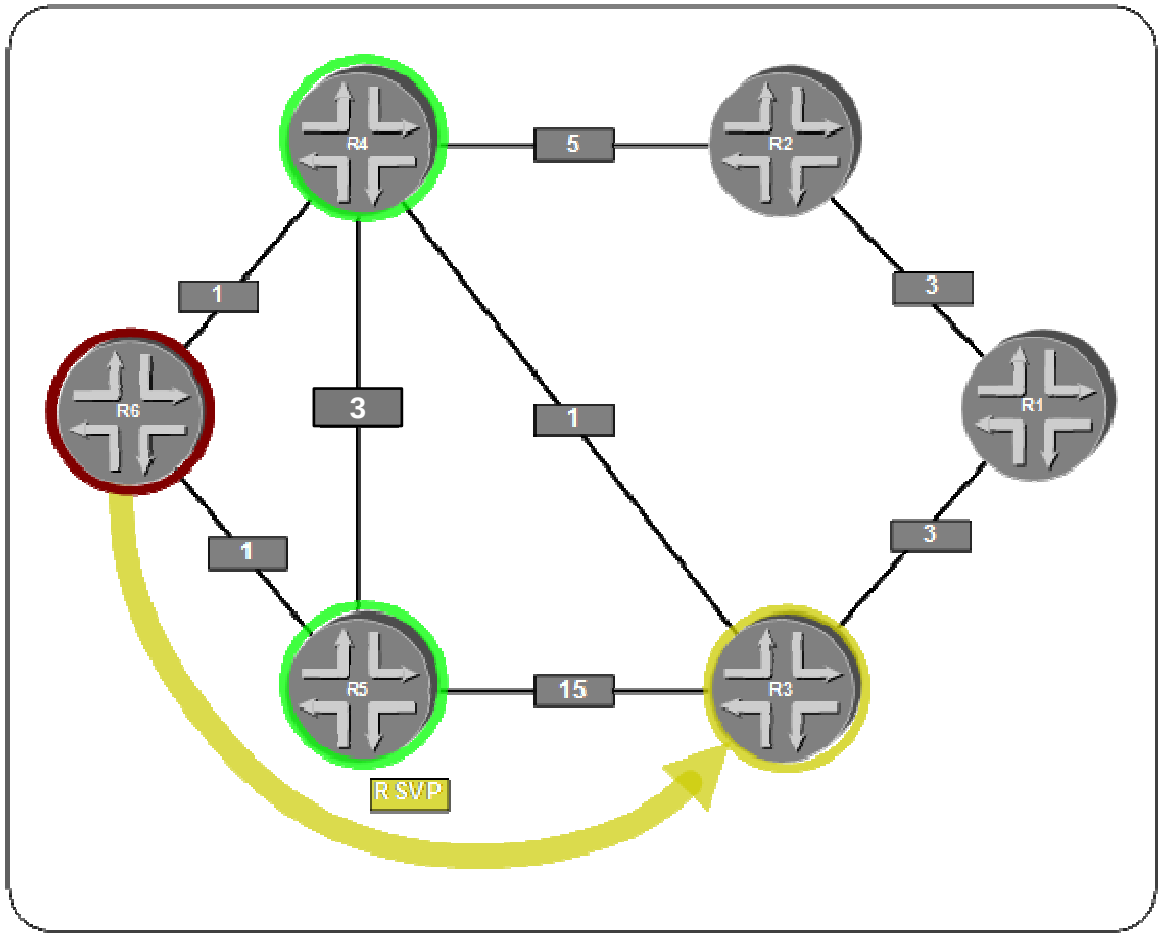


UI output

```
hannes@pro13-f> show isis backup spf results
pro13-h-lr1.00
  Primary next-hop: so-1/1/2.0, pro13-h
    Root: pro13-h, Metric: 20
    Root: pro13-e, Metric: 25
  Backup next-hop: fe-0/0/0.1001, pro13-e, SNPA:  0:19:e2:8a:b4:0

pro13-g-lr1.00, Address 0x8bede00
  Primary next-hop: so-1/1/2.0, pro13-h
    Root: pro13-h, Metric: 10
      Not eligible, Reason: Primary next-hop link fate sharing
    Root: pro13-e, Metric: 30
      track-item: pro13-h.00-00
      track-item: pro13-f.00-00
      Not eligible, Reason: Path loops
```

Backup Coverage extension

LFA coverage extension by MPLS tunnels



LFA coverage extension

- **Backup coverage typical at 65- 85 %**
  - **Add links**
  - **Add *backup* LSPs**
    - **MPLSP tail end is root for backup SPF**

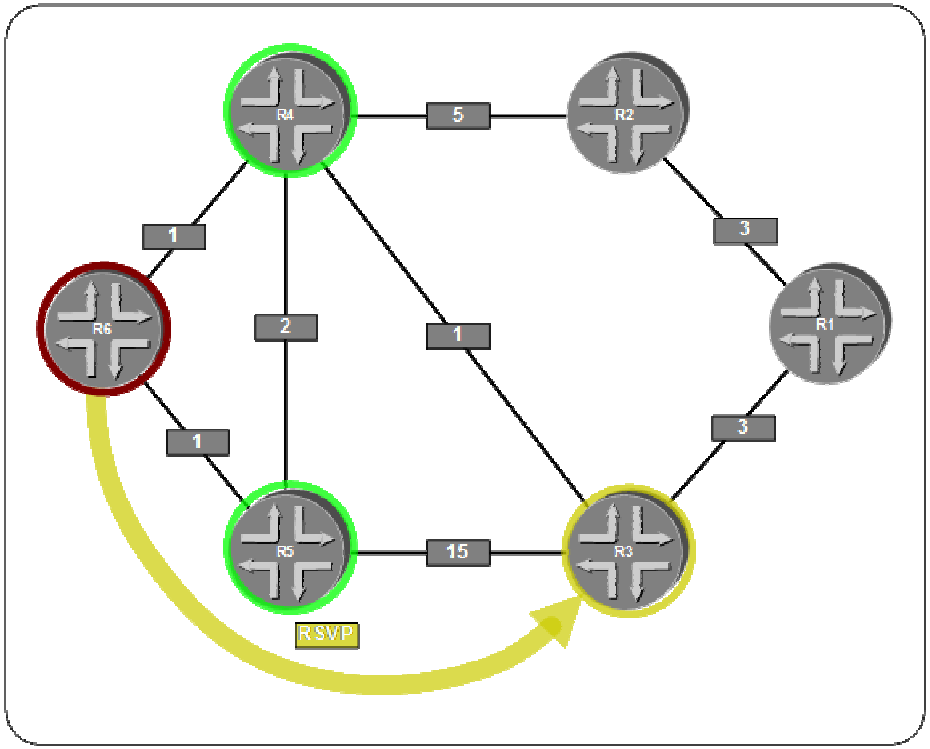
```
protocols {
    mpls {
        label-switched-path backup-rome {
            to 192.168.1.1;
            backup;
        }
    }
}
```

```
hannes@pro13-f> show isis backup label-switched-path
Backup MPLS LSPs:
f-to-g, Egress: 192.168.1.4, Status: up, Last change: 3d 22:42:33
  TE-metric: 24, Metric: 20, Refcount: 1
```

LDP integration

LDP integration

- Inherit next hop & weight from IGP tracking routes
- For ingress routes
- For transit routes
- Backup MPLSP requires ldp-tunneling



LDP integration

- ingress

```
hannes@pro13-f> show route 192.168.1.3 detail

192.168.1.3/32 (1 entry, 1 announced)
        State: <FlashAll>
        *LDP    Preference: 9
                Next hop type: Router
                Next-hop reference count: 2
                Next hop: via so-1/1/2.0 weight 0x1, selected
                Next hop: 10.0.1.1 via fe-0/0/0.1000 weight 0x4000
                Label operation: Push 299824
                State: <Active Int>
                Age: 3d 22:33:35        Metric: 1
                Task: LDP
                Announcement bits (1): 1-Resolve tree 1
                AS path: I
```

LDP integration

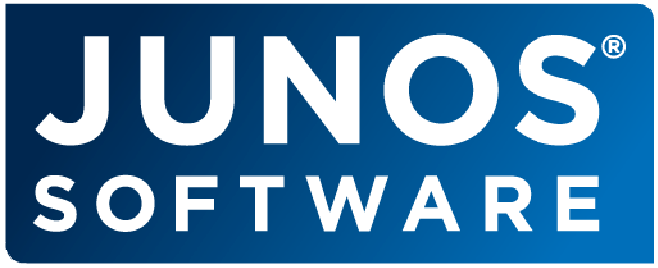
- **transit**

```
hannes@pro13-e> show route table mpls.0 detail

299856 (1 entry, 1 announced)
        *LDP    Preference: 9
                Next hop type: Router
                Next-hop reference count: 1
                Next hop: via so-1/1/2.0 weight 0x1, selected
                Label operation: Swap 299840
                Next hop: 10.0.1.2 via fe-0/0/0.1000 weight 0x4000
                Label operation: Swap 299841
                State: <Active Int>
                Local AS: 65535
                Age: 3d 22:37:19        Metric: 1
                Task: LDP
                Announcement bits (1): 0-KRT
                AS path: I
                Prefixes bound to route: 192.168.1.14/32
```

LFA Summary

- Protection of unicast in IP/MPLS network
- Reduces packetloss while routers are converging
- Adds fast-reroute FRR capabilities to ISIS, OSPF and LDP
- Rapid failure-repair via precalculated loop-free alternate next-hop
- No support from other routers needed
- Mixing of LFA and no LFA-enabled nodes allowed

Avaliability



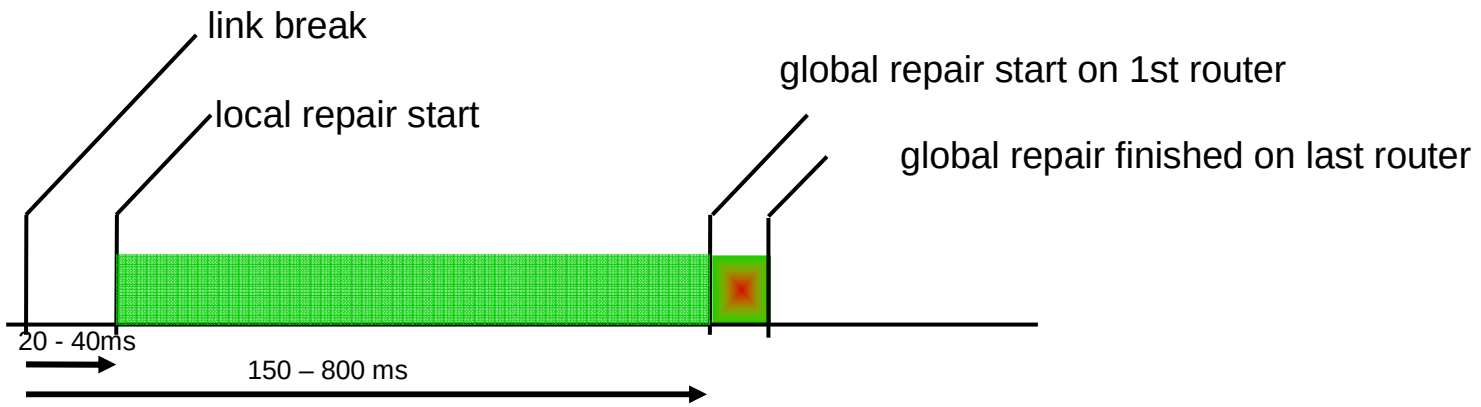
Loop-Free Alternates for ISIS already available in JUNOS release.

Loop-Free Alternates for OSPF will be available very soon.

Current JUNOS version is 9.6

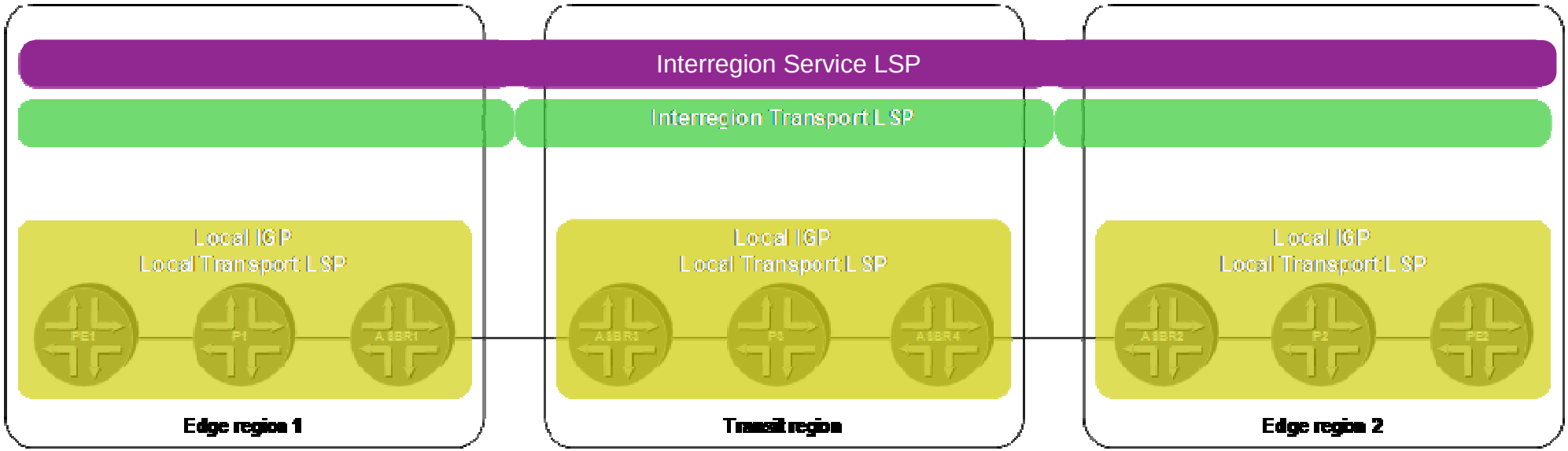
Caveats

Caveats, outstanding defects

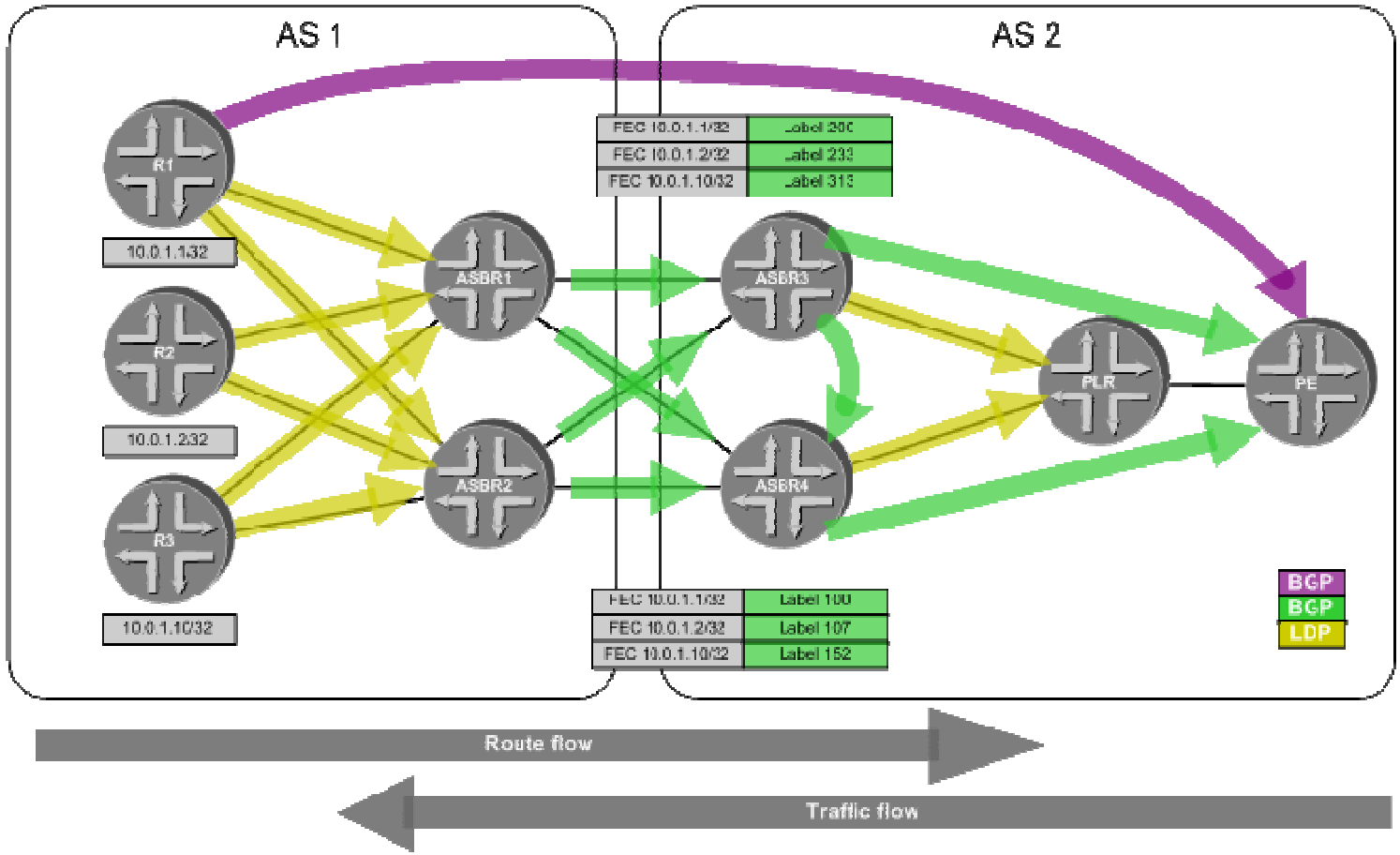


PROTECTING TUNNEL ENDS

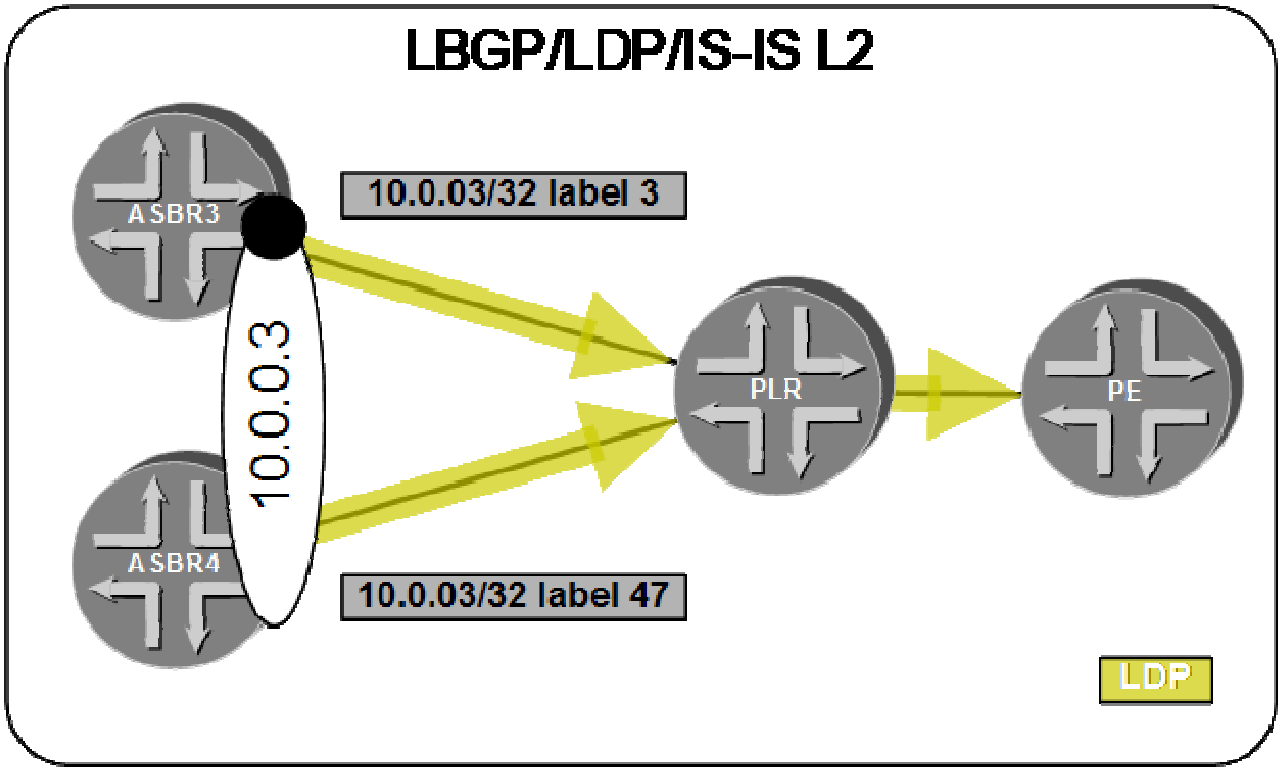
Generalized interdomain model inter-AS



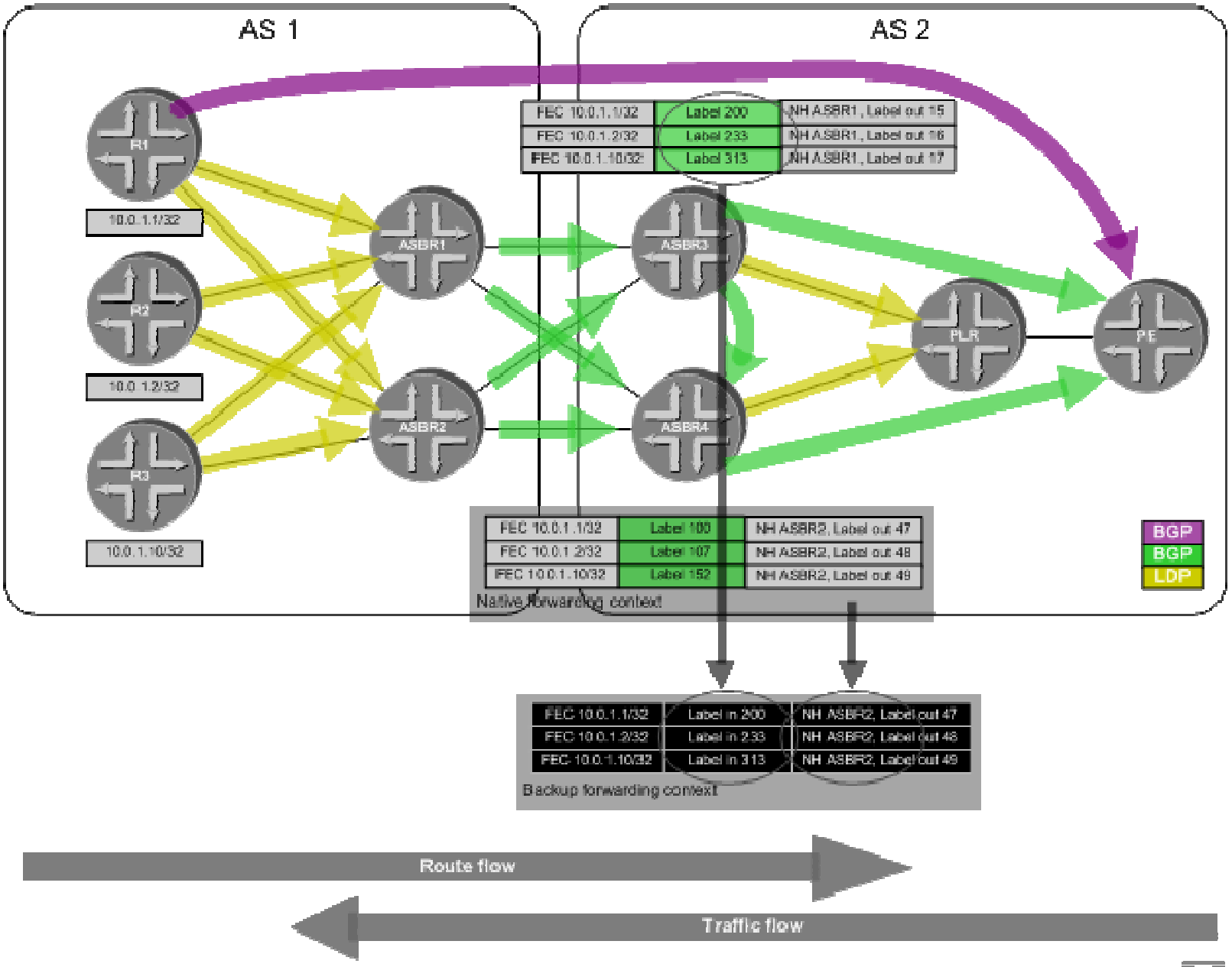
Local repair, how ?



Any casting BGP next-hops Example: LDP core transport



The big picture: Backup route splicing



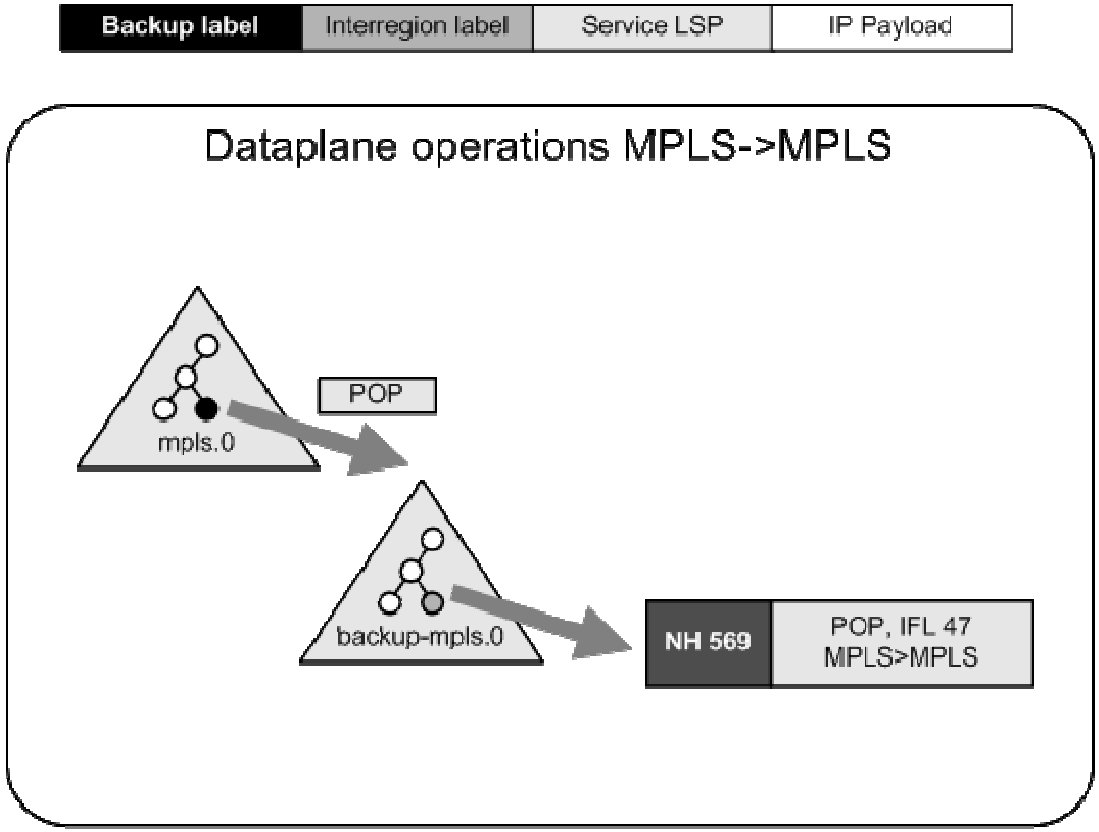
UI example: backup context generation

- No Protocol extensions required

```
protocols mpls {
    label-association {
        label-peer 192.168.1.3; ## ASBR3
        virtual-address 10.0.0.3;
    }
}
```

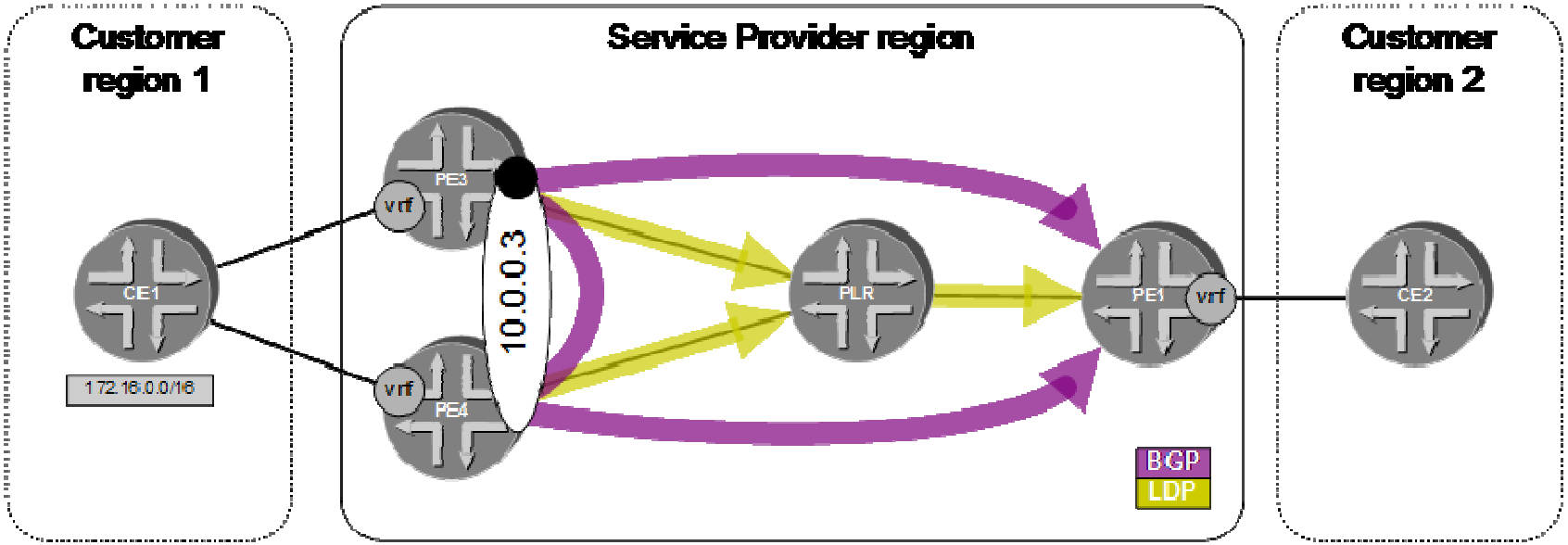
- Just box local behavior changes
  - Transport (context) label generation
  - Associate cloned FIBs with local FIB
    - No next-hop explosion / just MPLS route increase
  - Delay FIB update in case of failure

Transport LSP Data plane, ASBR4

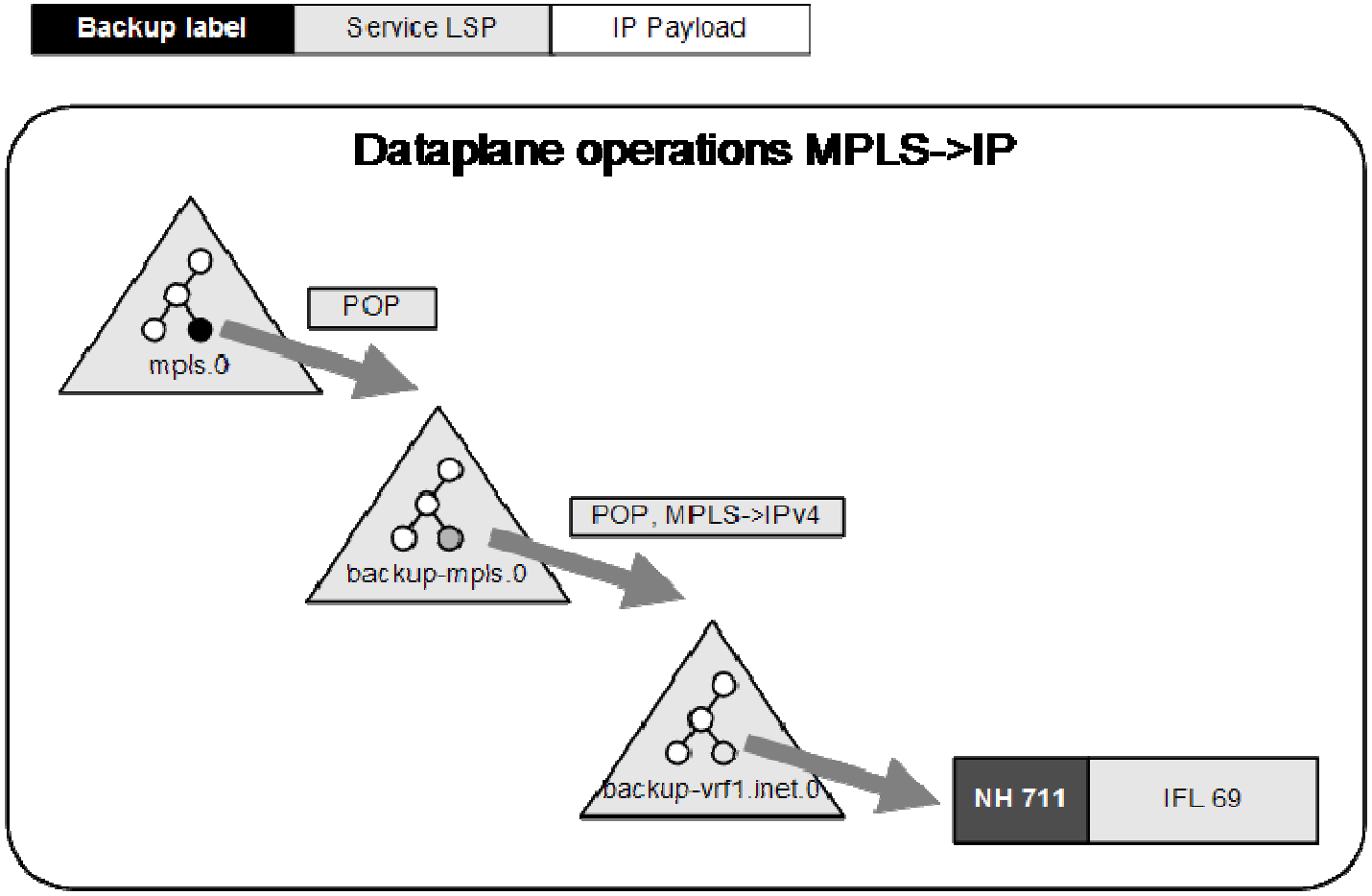


Example: 2547 service protection

Protecting a (Service) Tunnel endpoint

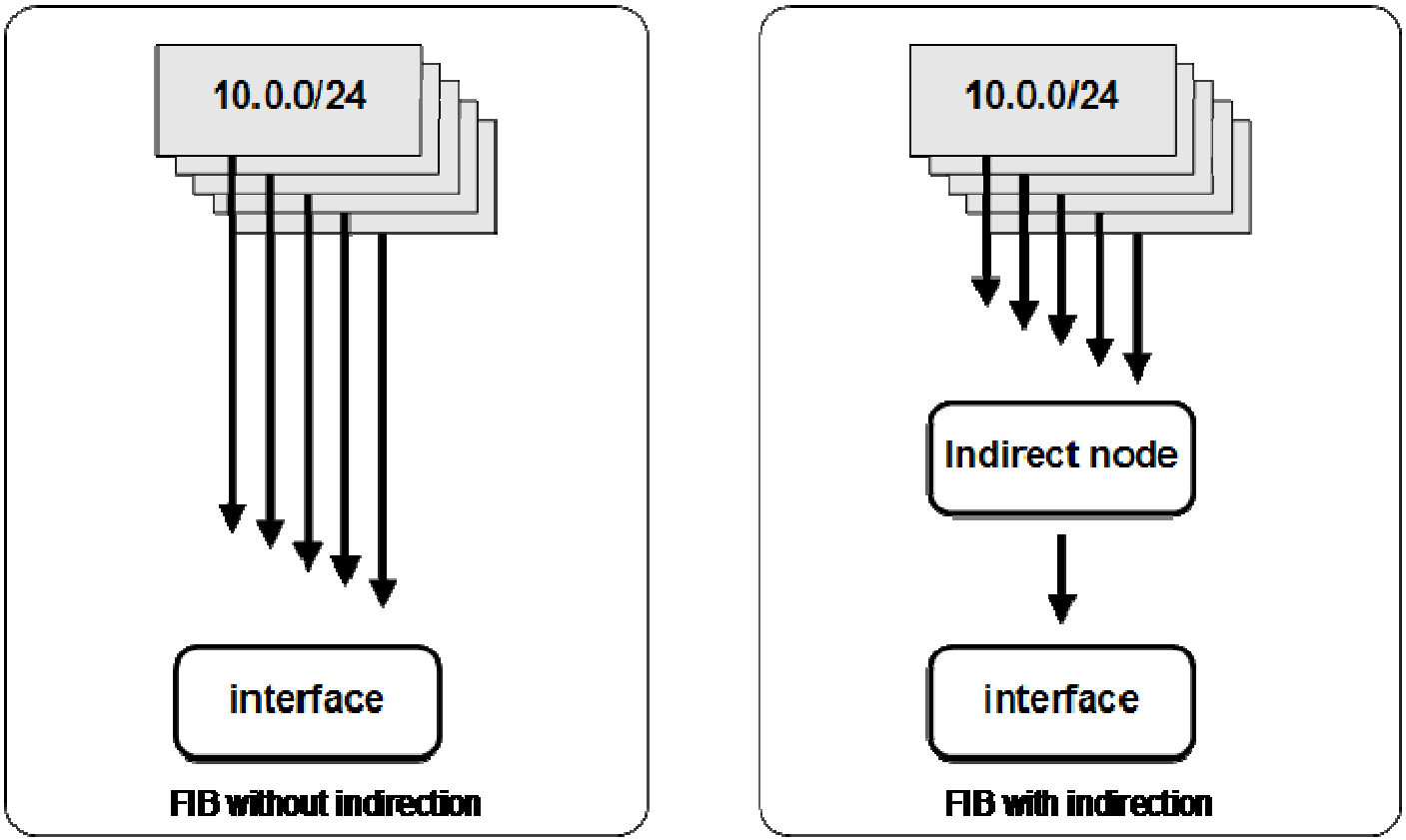


L3VPN Service LSP Data plane, ASBR4



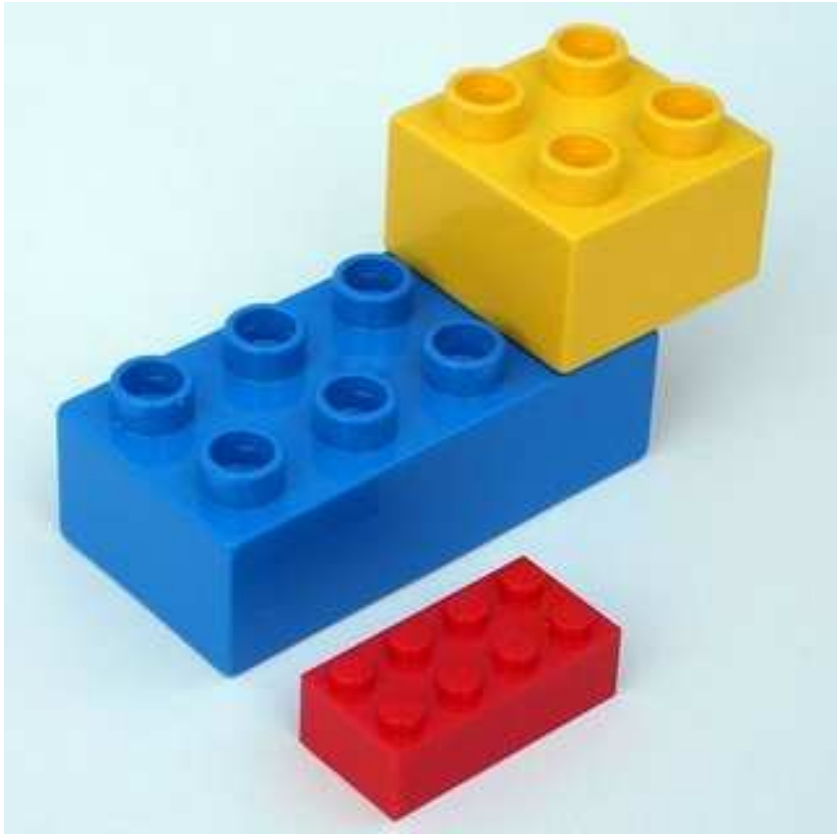
FIB Aggregation

FIB indirection



Summary

- **Fast convergence**
  - **does not require protocol extensions**
  - **Entirely a box local decision**
- **Solution applicable**
  - **Transport & service LSPs**
  - **Intra & interdomain boundaries**



Engineered for the network ahead™

THANK YOU