

Design & Implementation of zebra-rs

Single Process

Rust & tokio provides efficient resource management on multi core CPUs
tokio-console provide real-time performance monitoring

Unicast/LSP/SRv6

Native support of onlink nexthop, LSP and SRv6

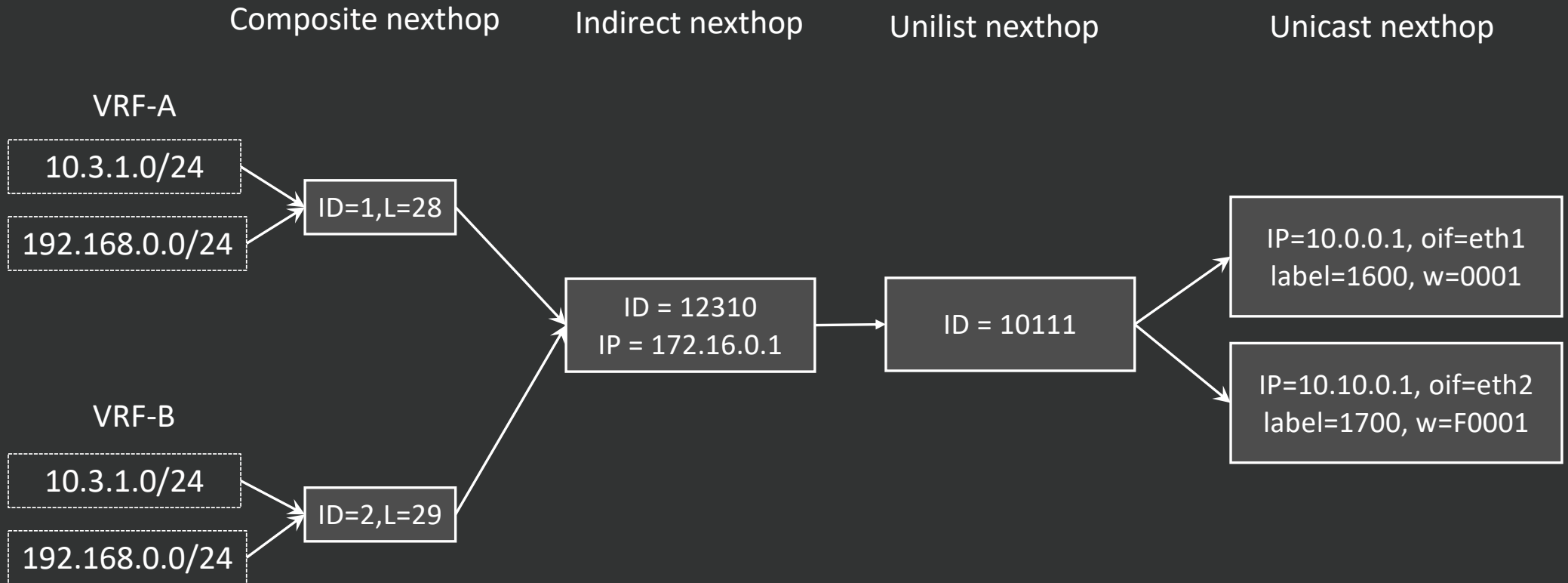
BGP/OSPF/IS-IS LDP

OSPFv2, OSPFv3 and IS-IS share common code using generics

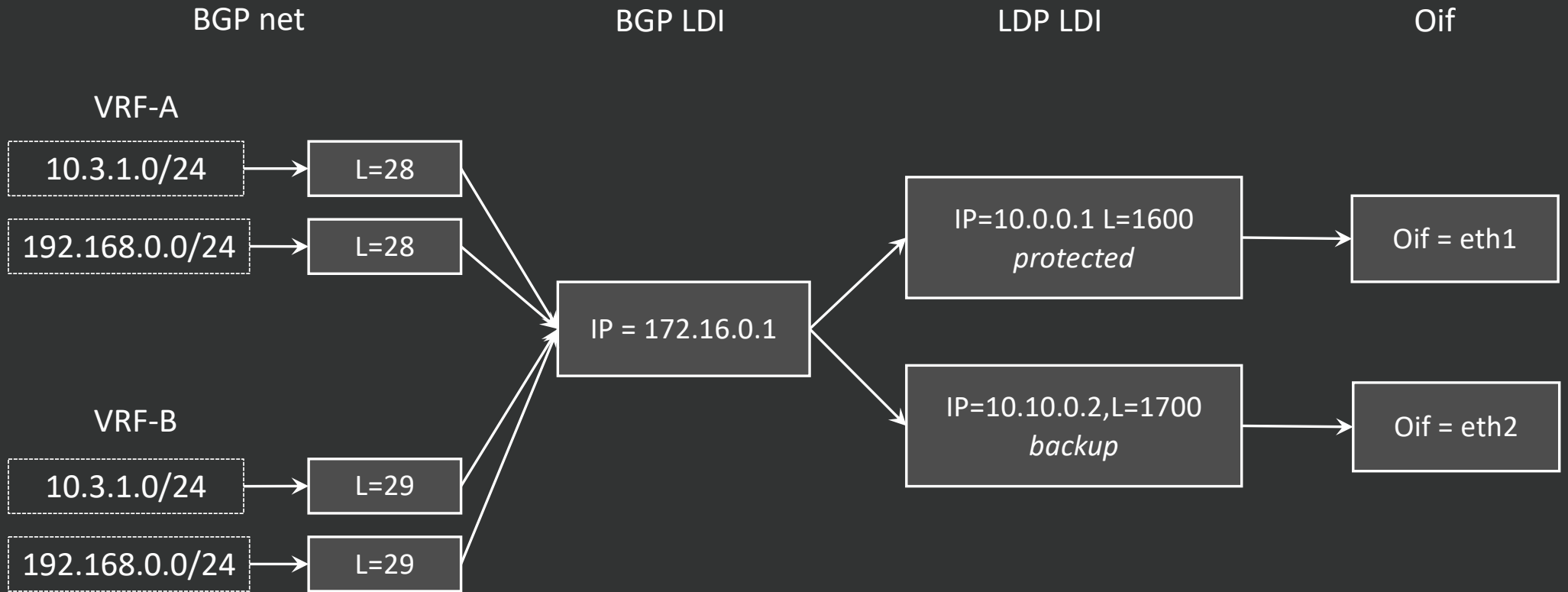
SD-WAN & Cloud Native

SD-WAN use case and reachability and forwarding decision separation
Cloud Native code with easy deployment

RIB Architecture – MPLS/VPN (Junos)



RIB Architecture – MPLS/VPN (IOS-XR)



LDI = Load Information

RIB Architecture – MPLS/VPN (Linux nexthop group)

Resolve nexthop
(in zebra-rs)

nexthop with encap

VRF-A

10.3.1.0/24

192.168.0.0/24

VRF = VRF-A
IP = 172.16.0.1
encap = 144

IP=10.0.0.1, oif=eth1 encap=16, 144, metric = 10

IP=10.10.0.2, oif=eth2 encap=17, 144, metric = 20

VRF-B

10.3.1.0/24

192.168.0.0/24

VRF = VRF-B
IP = 172.16.0.1
encap = 154

IP=10.0.0.1, oif=eth1 encap=16, 154, metric = 10

IP=10.10.0.2, oif=eth2 encap=17, 154, metric = 20

RIB Architecture – MPLS/VPN (Linux nexthop group)

```
$ ip nexthop
```

```
id 1020 via 10.2.0.2 dev enp0s7 scope link proto zebra  
id 1023 via 10.10.0.1 dev enp0s6 scope link proto zebra  
id 1024 encap mpls 16 via 10.10.0.1 dev enp0s6 scope link proto zebra  
id 1027 encap mpls 16/144 via 10.10.0.1 dev enp0s6 scope link proto zebra  
id 1028 encap mpls 16/154 via 10.10.0.1 dev enp0s6 scope link proto zebra
```

```
$ ip route show table 10
```

```
4.4.4.4 nhid 1027 encap mpls 16/144 via 10.10.0.1 dev enp0s6 proto bgp metric 20  
5.5.5.5 nhid 1020 via 10.2.0.2 dev enp0s7 proto ospf metric 20  
10.2.0.0/24 dev enp0s7 proto kernel scope link src 10.2.0.1  
local 10.2.0.1 dev enp0s7 proto kernel scope host src 10.2.0.1  
broadcast 10.2.0.255 dev enp0s7 proto kernel scope link src 10.2.0.1
```

RIB Architecture – Requirements

ECMP

Usually employs hash-based algorithms to ensure traffic is distributed evenly without packet reordering. Protocol: Static, OSPF, IS-IS, BGP

UCMP

Traffic is split proportionally, with higher traffic volume sent over lower-cost paths and less traffic over higher-cost paths. Protocol: Static, EIGRP

LSP Protection FRR & TI-LFA

Making Primary and Backup path is the typical way to protect LSP. We have a several way to achieve this. FRR is local repair protection. TI-LFA is loop free protection.

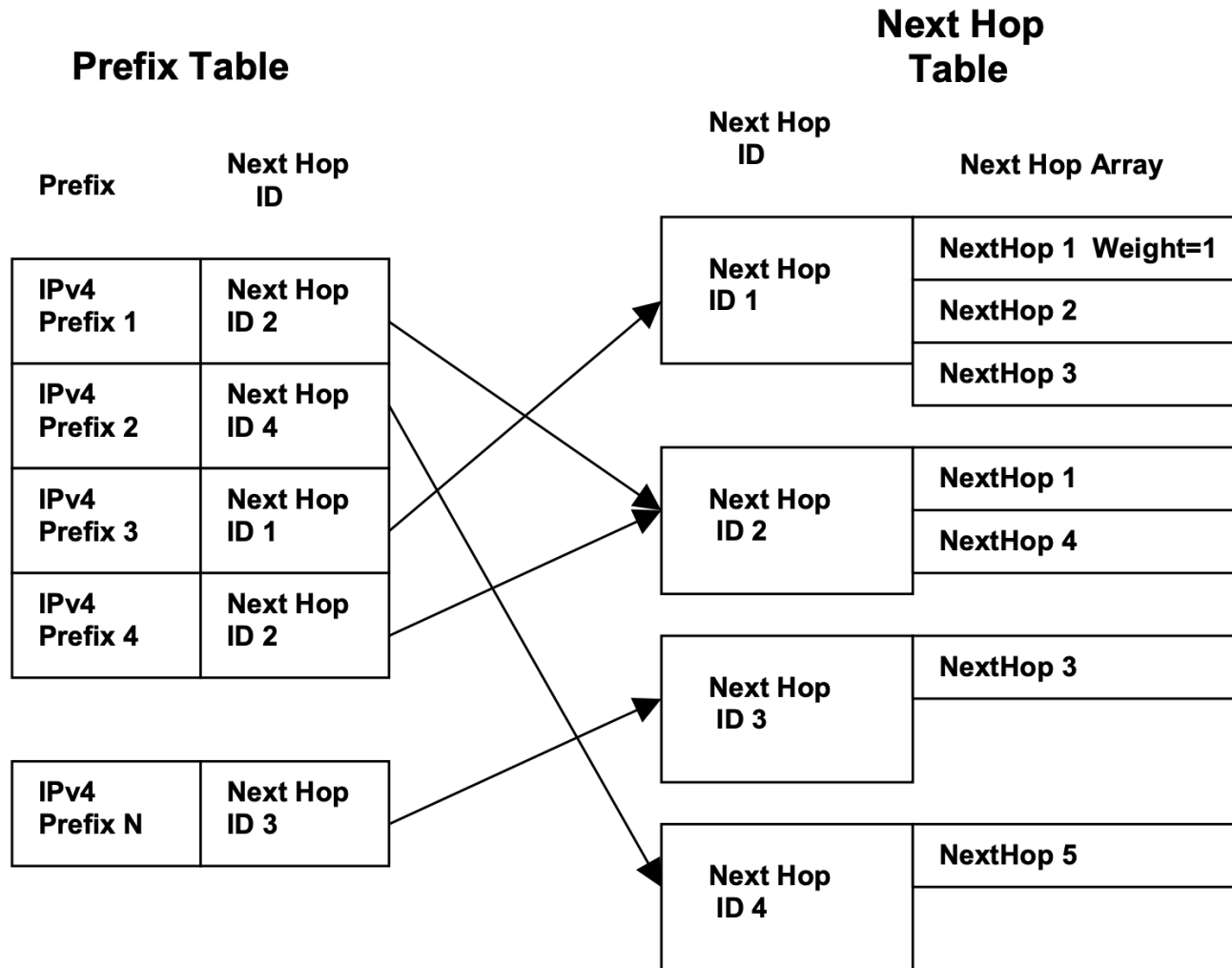
SD-WAN BGP AddPath

SD-WAN require multiple nexthop for same prefix for hybrid WAN services. Based on various criteria, forwarder pick up which nexthop to be used

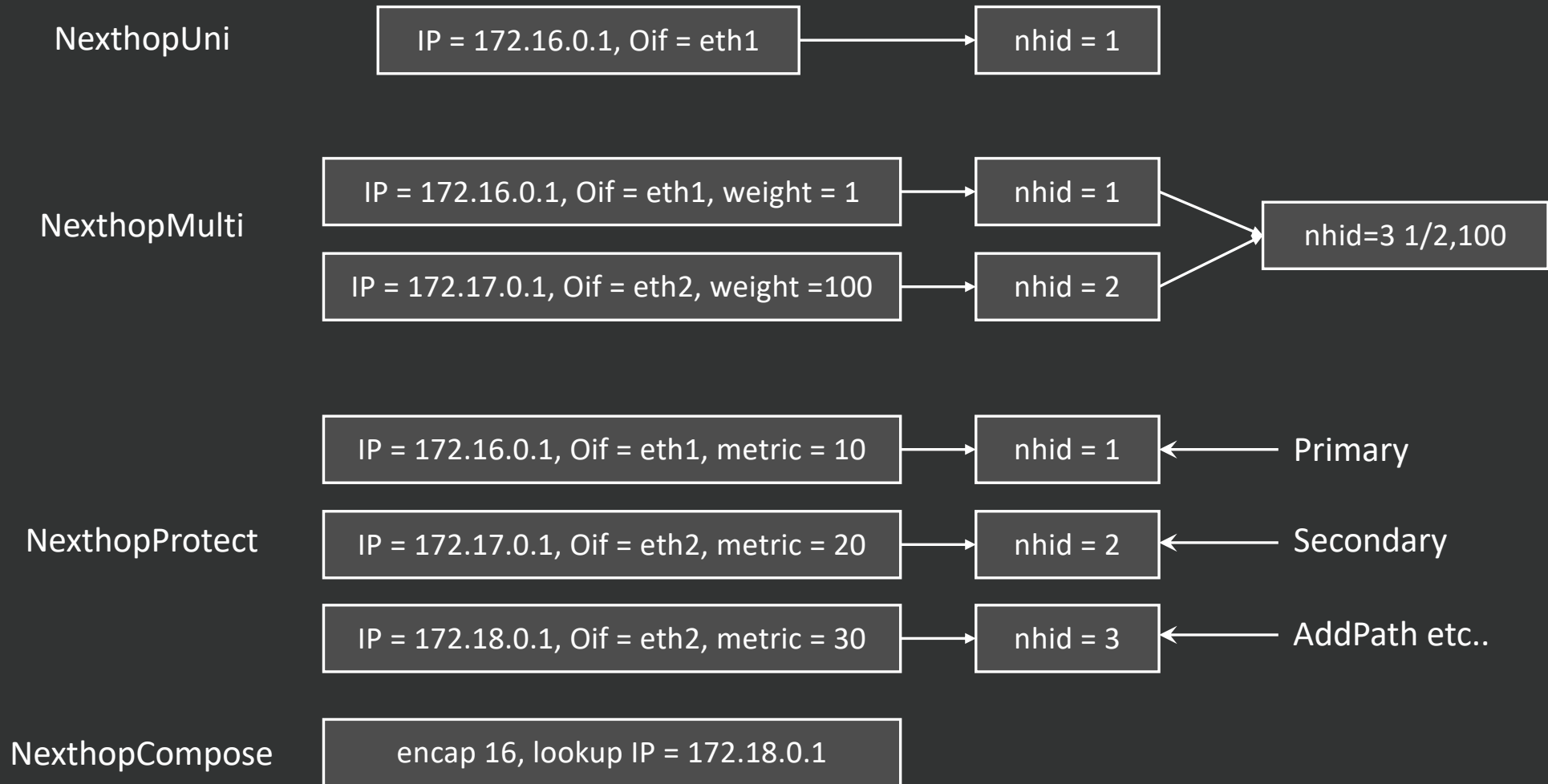
SD-WAN Local Breakout

Local breakout is the most common application in SD-WAN services. Based on service category, local system forward the packet through Default VRF's upstream default route

RIB Architecture – Network Processing Forum API (2004)



RIB Architecture – zebra-rs with Linux nexthop group



Management Architecture – CLI

• set/delete/show

```
macbookpro#show
routing-options {
  router-id 10.0.0.1;
}
macbookpro#delete routing-options
router-id

macbookpro#delete routing-options router-id
10.0.0.1

macbookpro#delete routing-options router-id 10.0.0.1
macbookpro#
```

• dynamic completion

```
MacBook-Pro.local>sh interfaces
+> <interface:string>
br-78dbd5b4a42f
br-f5be5e13a83c
docker0
enp0s5
enp0s6
lo

MacBook-Pro.local>sh interfaces enp
enp0s5 enp0s6
MacBook-Pro.local>sh interfaces enp0s5
Interface: enp0s5
Hardware is Ethernet<macaddress>
```

• remote access

```
$ ~ export VTYSH_SERVER_URL="http://10.211.55.91"
$ ~ vtysh
MacBook-Pro.local>sh ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area, N1/N2 - OSPF NSSA external type 1/2
       E1/E2 - OSPF external type 1/2 D - DHCP route
       i - IS-IS, L1/L2 - IS-IS level-1/2, ia - IS-IS inter area
       > - selected route, * - FIB route, S - Stale route

D    *> 0.0.0.0/0 [0/100] via 10.211.55.1, enp0s5
S    *> 1.1.1.1/32 [1/0] via 10.211.55.1, enp0s5, metric 0
      via 10.211.55.2, enp0s5, metric 10
C    *> 10.211.55.0/24 directly connected enp0s5
C    *> 169.254.0.0/16 directly connected br-78dbd5b4a42f
```

• real shell

```
MacBook-Pro.local#json
{"routing":{"static":{"ipv4":{"route": [{"prefix":"1.1.1.1/32","nexthop": [{"ress":"10.211.55.1"}, {"address":"10.211.55.2","metric":"10"}]}]}}}}
MacBook-Pro.local#json | jq .
{
  "routing": {
    "static": {
      "ipv4": {
        "route": [
          {
            "prefix": "1.1.1.1/32",
            "nexthop": [
              {
                "address": "10.211.55.1"
              }
            ]
          }
        ]
      }
    }
  }
}
```

Management Architecture – Config

Junos

```
routing {
  bgp {
    global {
      as 1;
      identifier 10.0.0.1;
    }
    neighbors {
      neighbor 10.0.0.2 {
        peer-as 1;
      }
      neighbor 10.0.0.3 {
        peer-as 1;
      }
    }
  }
  static {
    ipv4 {
      route 1.1.1.1/32 {
        nexthop 10.211.55.1;
        nexthop 10.211.55.2 {
          metric 10;
        }
      }
    }
  }
}
ubuntu#
```

JSON

```
{
  "routing": {
    "bgp": {
      "global": {
        "as": "1",
        "identifier": "10.0.0.1"
      },
      "neighbors": {
        "neighbor": [
          {
            "remote-address": "10.0.0.2",
            "peer-as": "1"
          },
          {
            "remote-address": "10.0.0.3",
            "peer-as": "1"
          }
        ]
      }
    },
    "static": {
      "ipv4": {
        "route": [
          {
            "prefix": "1.1.1.1/32",
            "nexthop": [
              {
                "address": "10.211.55.1",
                "metric": 10
              },
              {
                "address": "10.211.55.2",
                "metric": 10
              }
            ]
          }
        ]
      }
    }
  }
}
```

YAML

```
routing:
  bgp:
    global:
      as: '1'
      identifier: 10.0.0.1
    neighbors:
      neighbor:
        - remote-address: 10.0.0.2
          peer-as: '1'
        - remote-address: 10.0.0.3
          peer-as: '1'
  static:
    ipv4:
      route:
        - prefix: 1.1.1.1/32
          nexthop:
            - address: 10.211.55.1
              metric: '10'
            - address: 10.211.55.2
              metric: '10'
```

Management Architecture – K8S Culture



kubernetes

kubectl apply

```
routing:
  bgp:
    global:
      as: '1'
      identifier: 10.0.0.1
    neighbors:
      neighbor:
      - remote-address: 10.0.0.2
        peer-as: '1'
      - remote-address: 10.0.0.3
        peer-as: '1'
  static:
    ipv4:
      route:
      - prefix: 1.1.1.1/32
        nexthop:
        - address: 10.211.55.1
        - address: 10.211.55.2
        metric: '10'
```

```
$ ~ vtyctl apply -f bgp.yaml -h 10.211.55.91
$ ~ vtysh
ubuntu>configure
ubuntu#show
routing {
  bgp {
    global {
      as 1;
      identifier 10.0.0.1;
    }
    neighbors {
      neighbor 10.0.0.2 {
        peer-as 1;
      }
      neighbor 10.0.0.3 {
        peer-as 1;
      }
    }
  }
  static {
    ipv4 {
      route 1.1.1.1/32 {
        nexthop 10.211.55.1;
        nexthop 10.211.55.2 {
          metric 10;
        }
      }
    }
  }
}
```