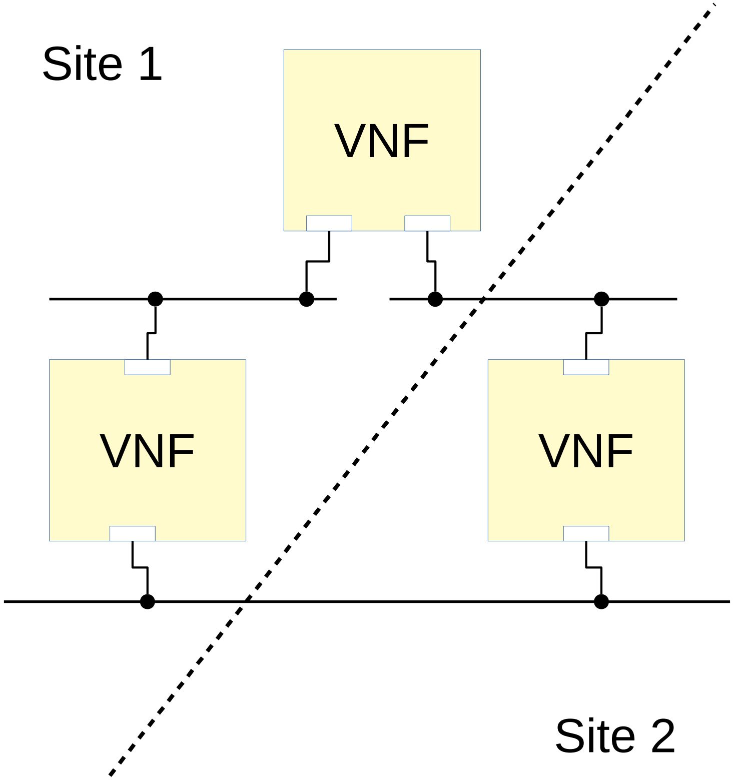
Dataplane Broker (DPB)

Steven Simpson, Arsham Farshad, Paul McCherry, Abubakr “Ali” Magzoub

Problem statement

- Multi-site (multi-VIM)
  - Each VNF assigned to a site
  - Some VLs split across sites
  - WIM responsible for inter-site connectivity
- Dataplane Broker (DPB)
  - Can act as WIM



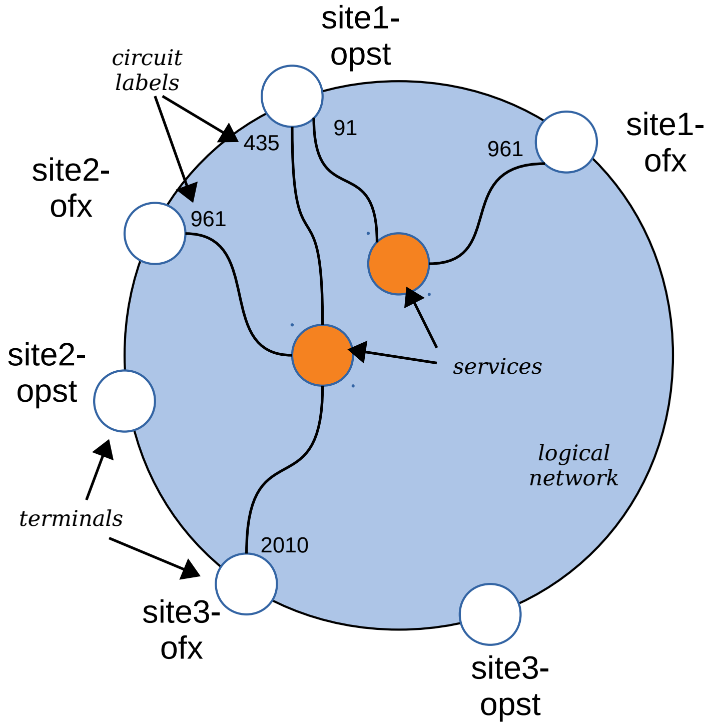
Wide-area L2 connections

- VLAN endpoints
  - Functional isolation of VLs
- Multipoint
  - NSes can be split over 2+ sites
- Bandwidth guarantees
  - Non-functional isolation
  - Traffic from one NS shouldn’t be able to drown out another
  - Asymmetric
- Multiswitch
  - Plugin framework for base ‘fabric’ layer
  - Heterogeneous physical network
    - Corsa DP2000 series
    - Generic OpenFlow
- Scalability
  - Hierarchical abstraction
  - Not looking for optimal solution
- OpenSource

Network abstraction

- Named terminals
  - Associated with sliced resources at specific locations, e.g., lancaster-openstack, paris-vpngw, berlin-ofx
- Numerically labeled circuits
  - Distinguishes services occupying same terminal
  - Maps to encapsulation technology (e.g., VLAN ids)
- Services
  - Connect 2+ circuits
  - Bandwidth guarantees

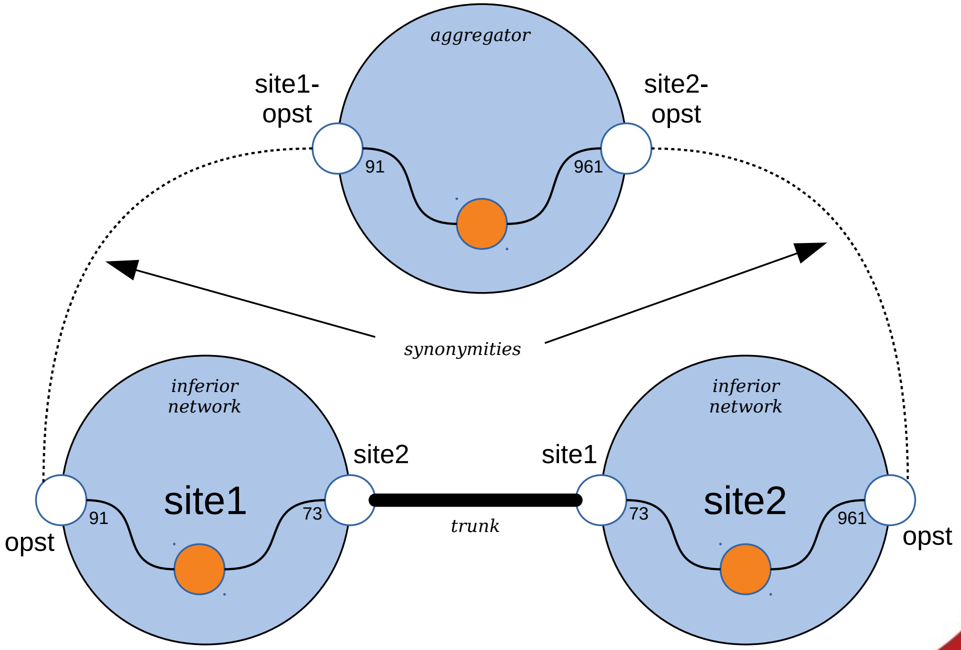
- Logical switch
  - Logical network subtype
  - Maps directly to physical switch
  - Uses adaptor to map to fabric technology



Aggregator

- Control of inferior networks
  - ‘Trunks’ connect ‘internal’ terminals of inferiors
  - Own terminals map to ‘external’ terminals of inferiors
  - Aggregator manages capacity of its own trunks
  - Aggregator service maps to set of inferior services

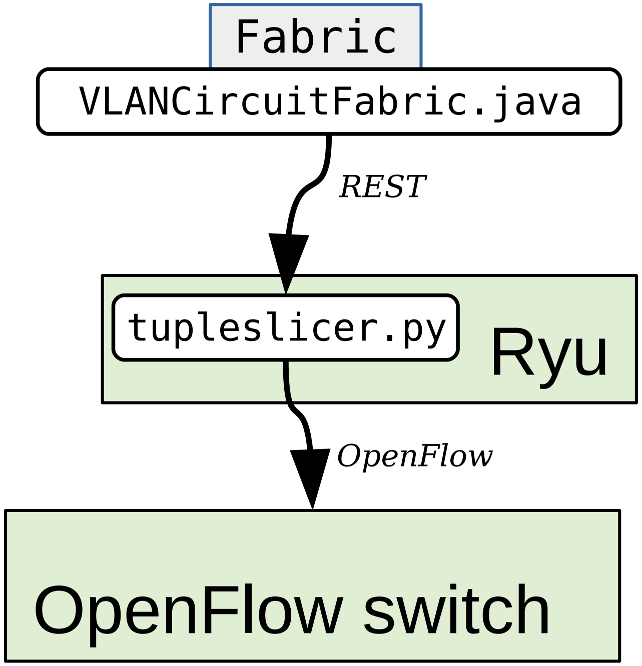
- Same northbound interface
  - Hierarchies could be built
  - Inferiors are either more aggregators, or ‘logical switches’
  - Leaves are always switches



Fabric adaptation

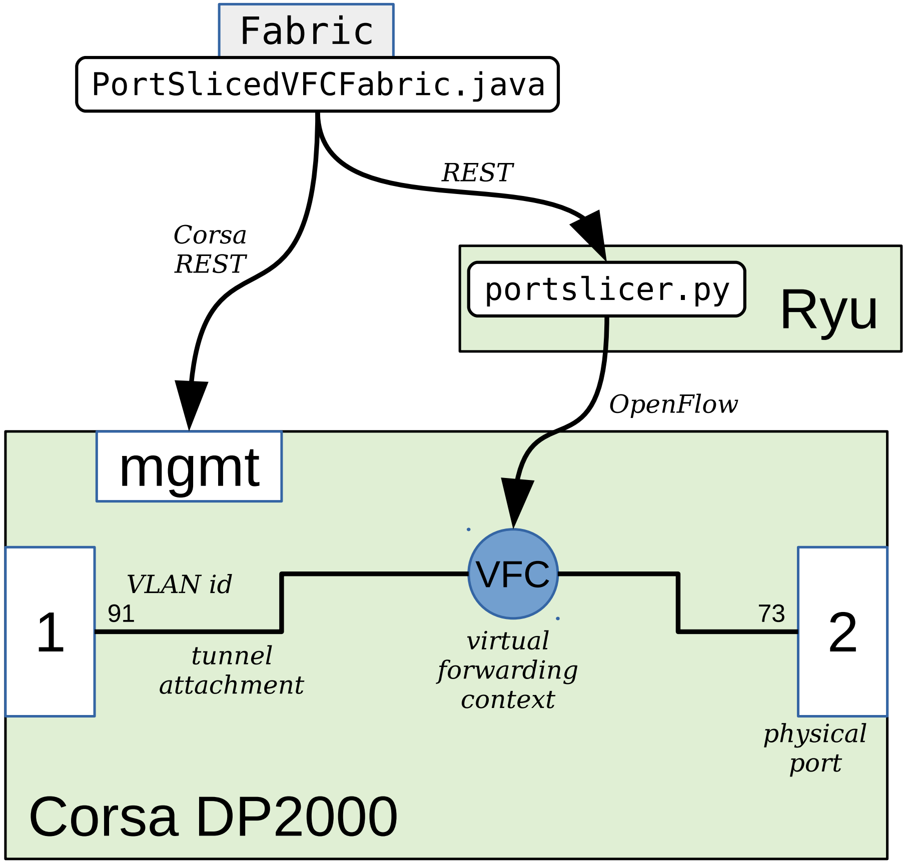
- OpenFlow adaptor
  - Uses VLAN OF operations for VLAN switching
  - Some metering applied to implement QoS
    - OF1.5
  - Custom Ryu controller app implements multiple isolated learning switches in one physical switch

- Fabric adaptors are plugins for specific technologies
  - Different adaptor usable by each logical switch
  - Network heterogeneity
  - No persistent state

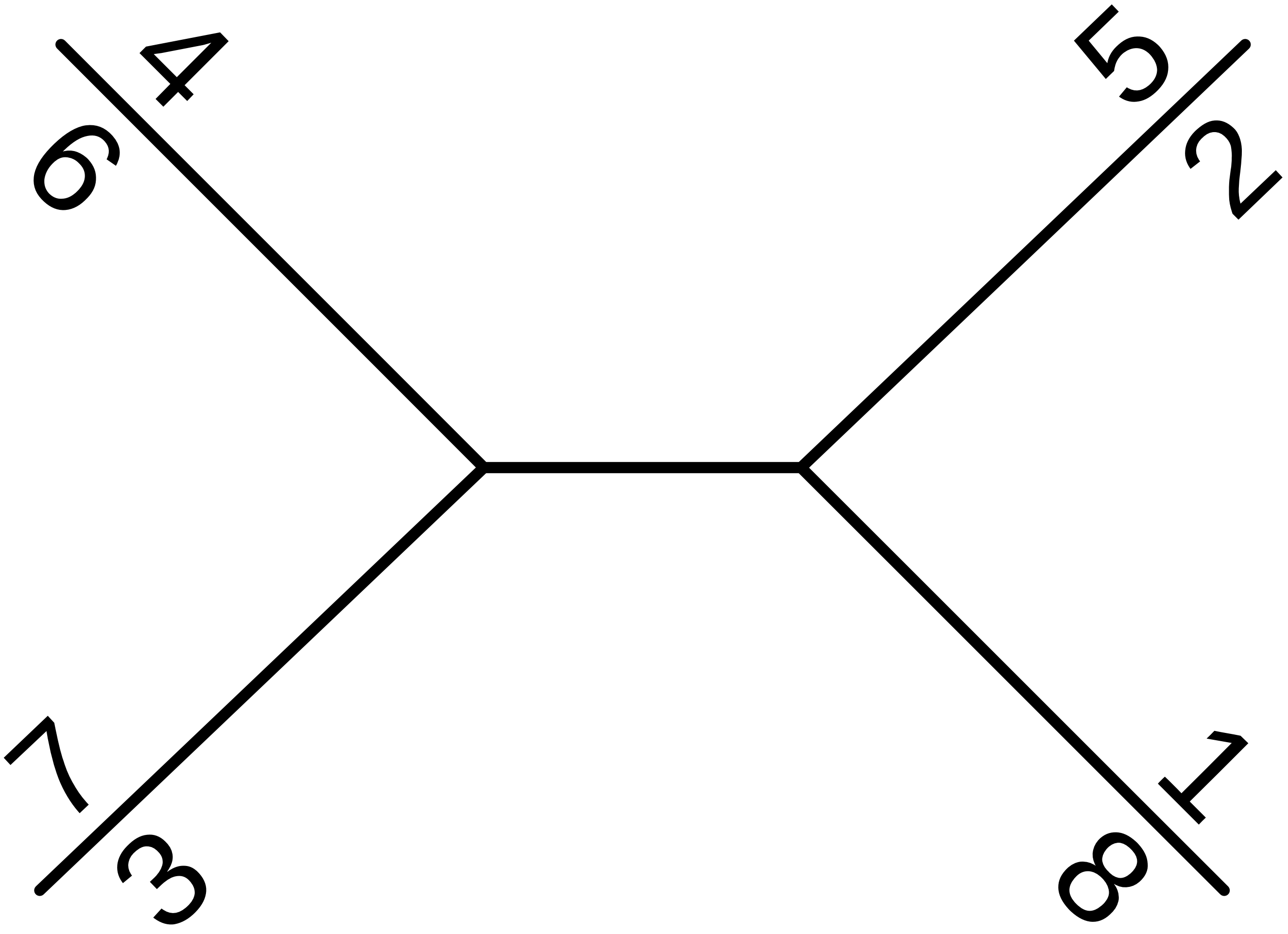


Fabric adaptation

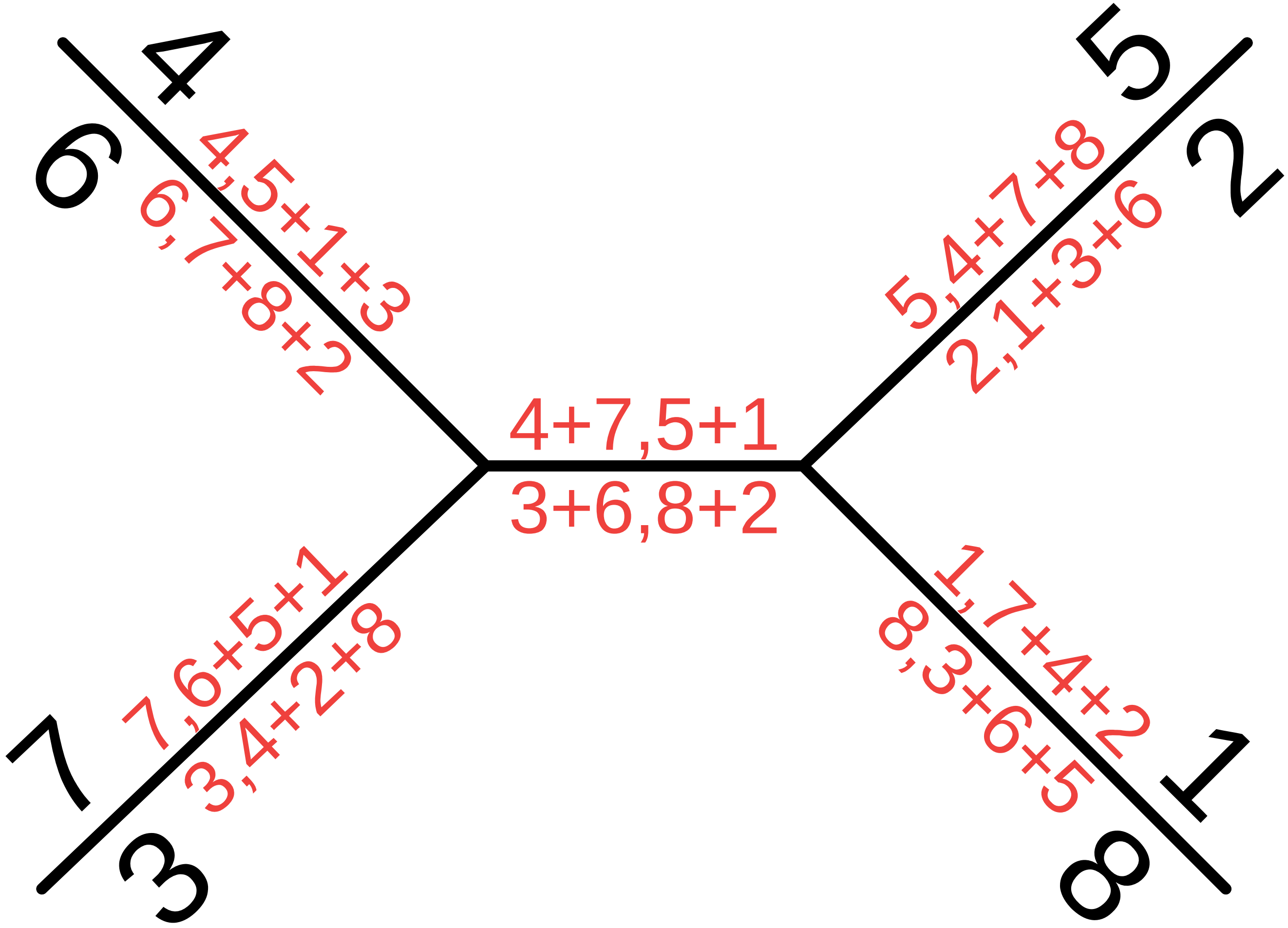
- Corsa adaptor
  - Uses custom Ryu app to switch between internal ports of VFC
  - Uses switch management REST API to attach VFC ports to physical ports and VLANs
    - (De-)tagging handled by attachments, not by OpenFlow
    - Shaping applied to attachments
      - QoS not implemented by OpenFlow



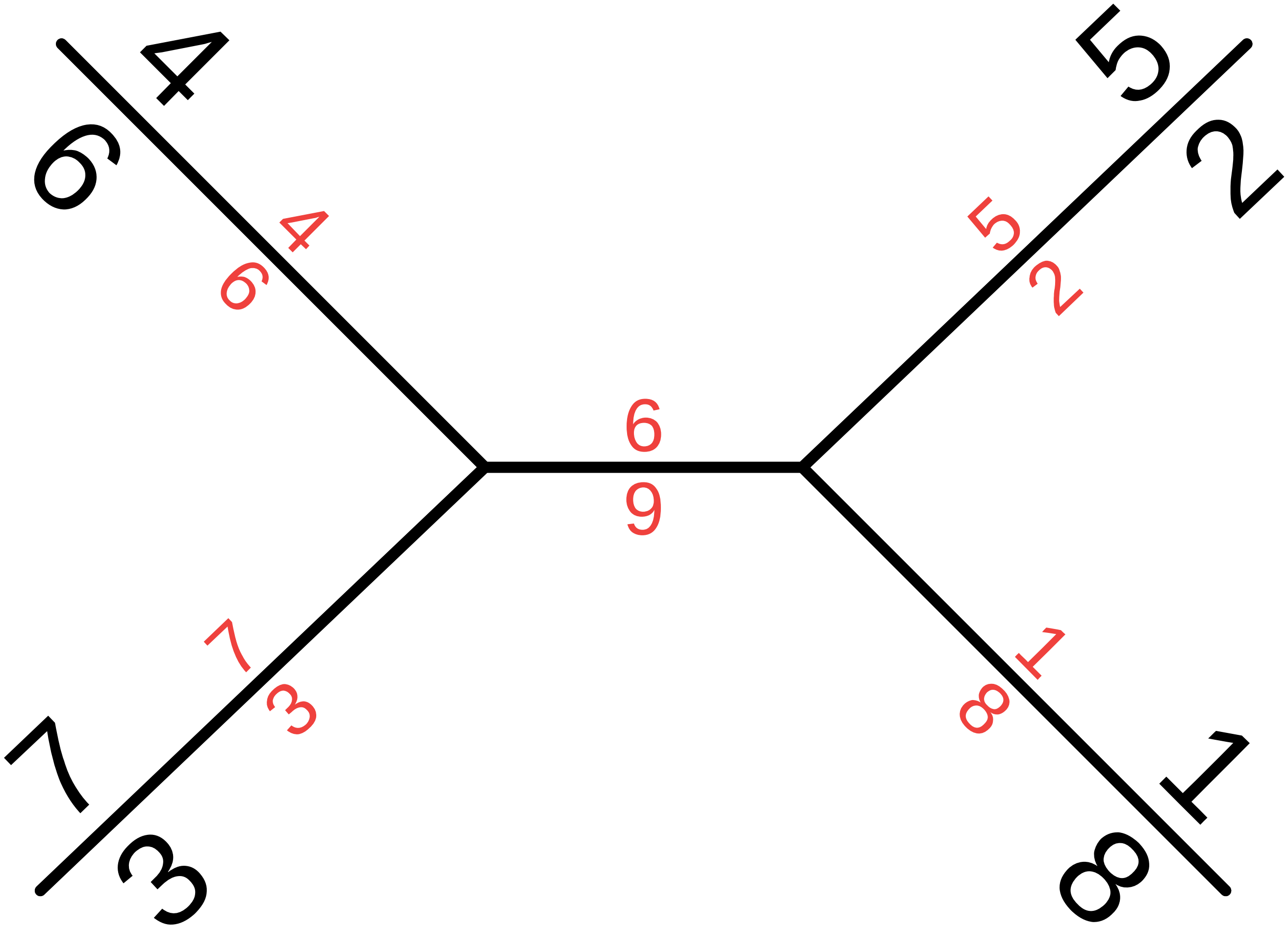
Aggregate bandwidths



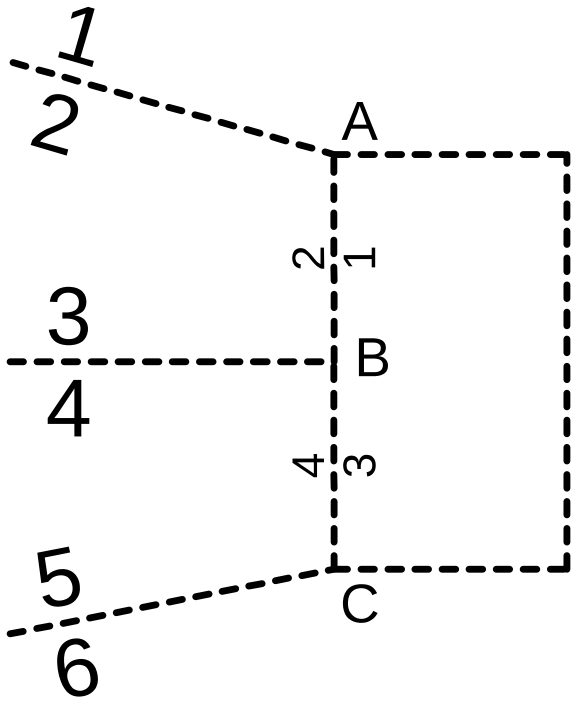
Aggregate bandwidths

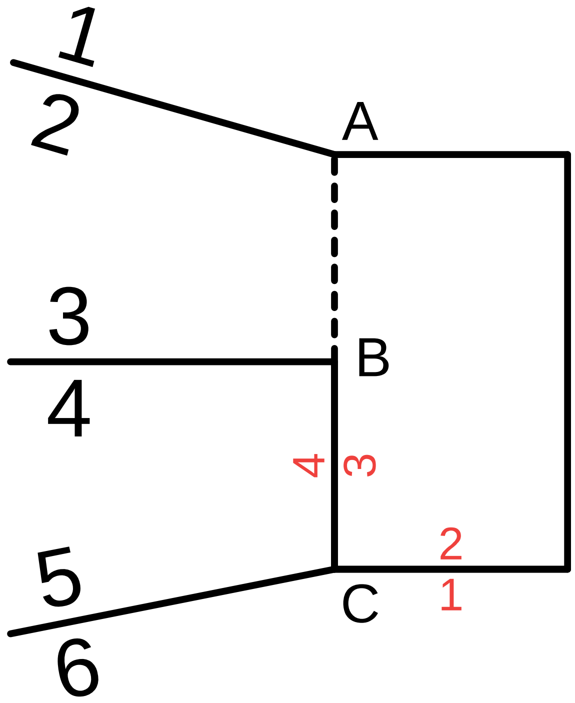


Aggregate bandwidths

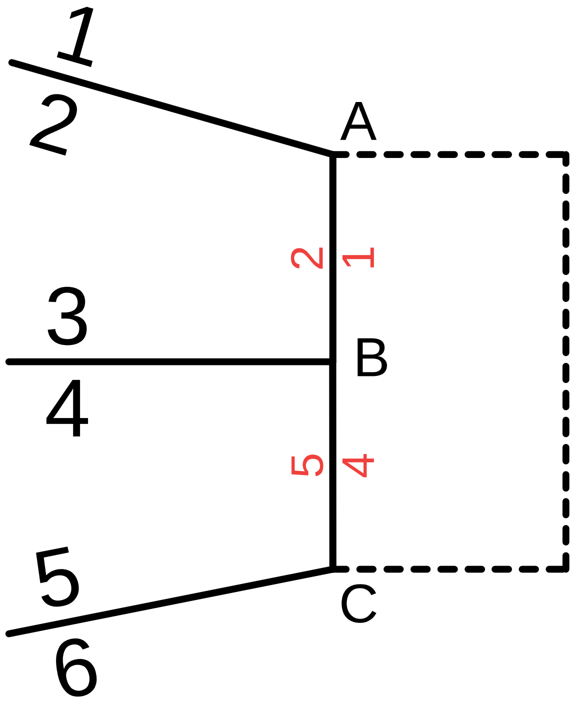


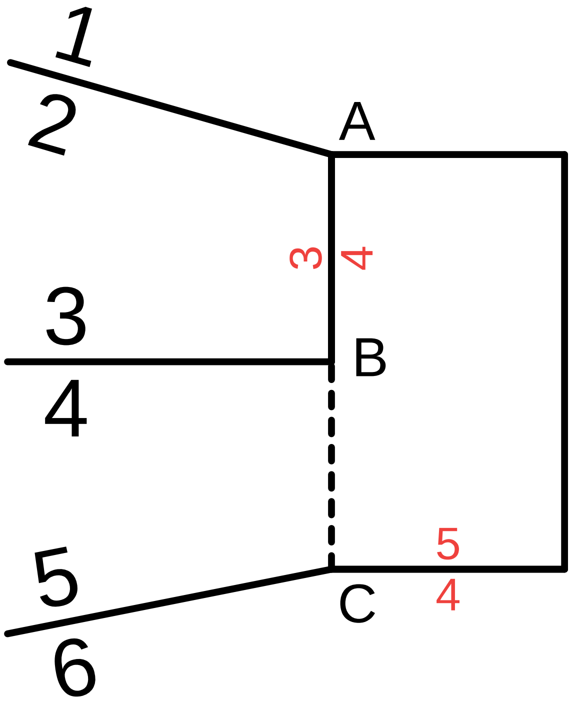
Sub-optimal results





Sub-optimal results





Future of DPB

- Service modification
  - Pretend that resources consumed by current configuration are available for new
- Bandwidth matrix
  - For better expression of (say) E-TREE
- OVSDB as fabric
  - Similar to Corsa architecture
- Multi-segment
  - Establish all disjoint segments or fail
- Alternative metrics for path computation
  - Latency, reliability, …
- Multitenancy
  - In the control plane
  - Better isolation of one user's services from other users' control

Acknowledgements

INITIATE

EPSRC
Engineering and Physical Sciences
Research Council

TOUCAN
ULTIMATE NETWORK CONVERGENCE

OSM multi-VIM issues

- IP pool splitting
  - OSM must co-ordinate IP configuration as it splits VL, not after
  - Same subnet; disjoint IP pools
  - Our work-around: block DHCP
  - Watch out for connected internal and external VLDs
  - What about switch-like and router-like behaviour across interfaces?
  - Holistic solution to related issues?

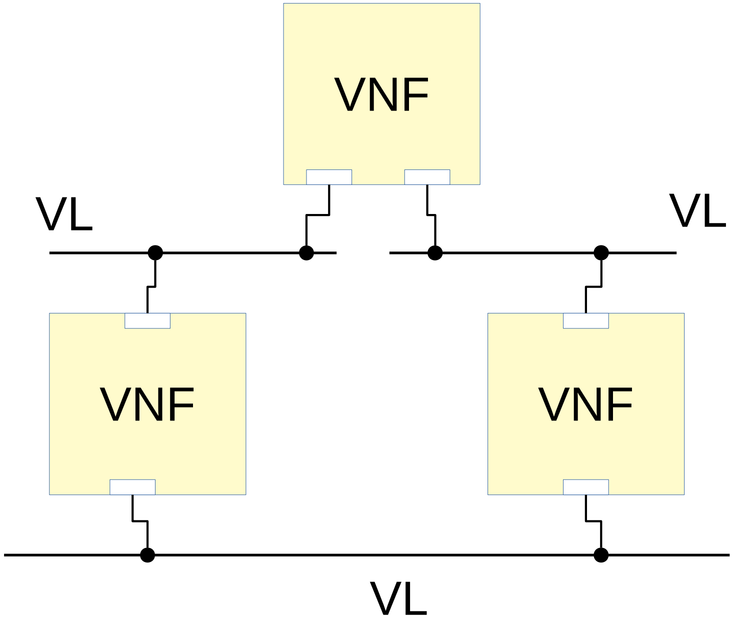
- Pre-existing networks
  - (including management)
  - Don't connect them during ns-create!
  - Assume they are already connected
  - Or deal with:
    - Modification of existing services
    - Merging of two services into one
    - Surprise unrelated subnets
  - Detection:
    - vim-network-name expressed or implied; and
    - profile unspecified

Multi-tenant multi-VIM management networks

- Per-tenant VIM configurations
  - Distinct VIM tenants and default management network names
  - Per-tenant isolation of management networks
  - Overlapping subnets
  - Juju client needs distinct netns context to access multiple simultaneously
    - VPN in?

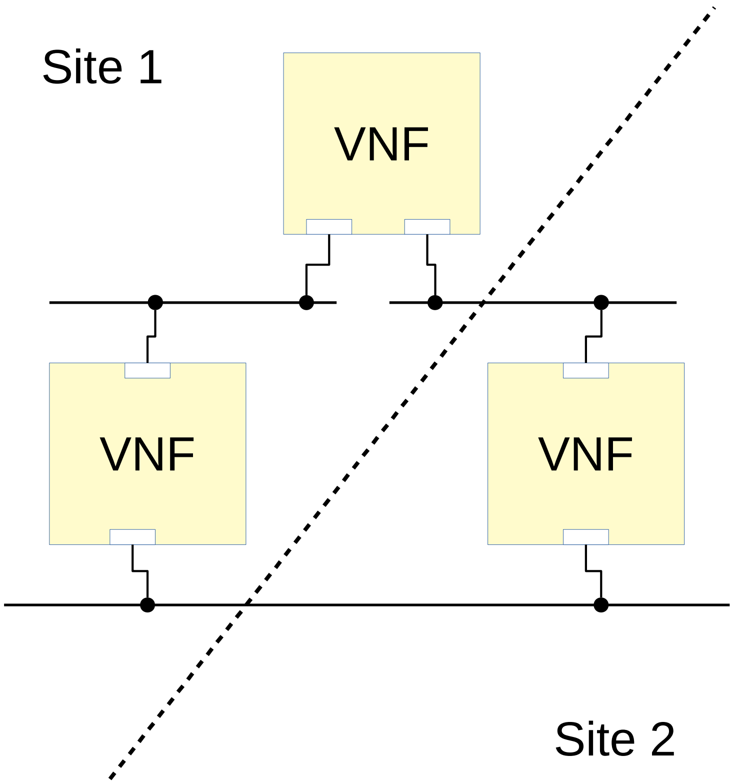
- Tool to set up multi-VIM management network?
  - Admin credentials of OSM and all requested VIMs
  - Create VIM projects at each site
    - Create VIM network
  - Create VPN gateway(s)
    - vpnmgr
  - Gather endpoints and connect with broker
  - Create OSM tenant
    - Populate with VIMs' project credentials and local network names
    - Provide Juju with VPN credentials
- Or do it through OSM?
  - Need VPN gateways as VNFs
  - Need VLD pinning (or dummy VNFs)

Multi-VIM IP pool split



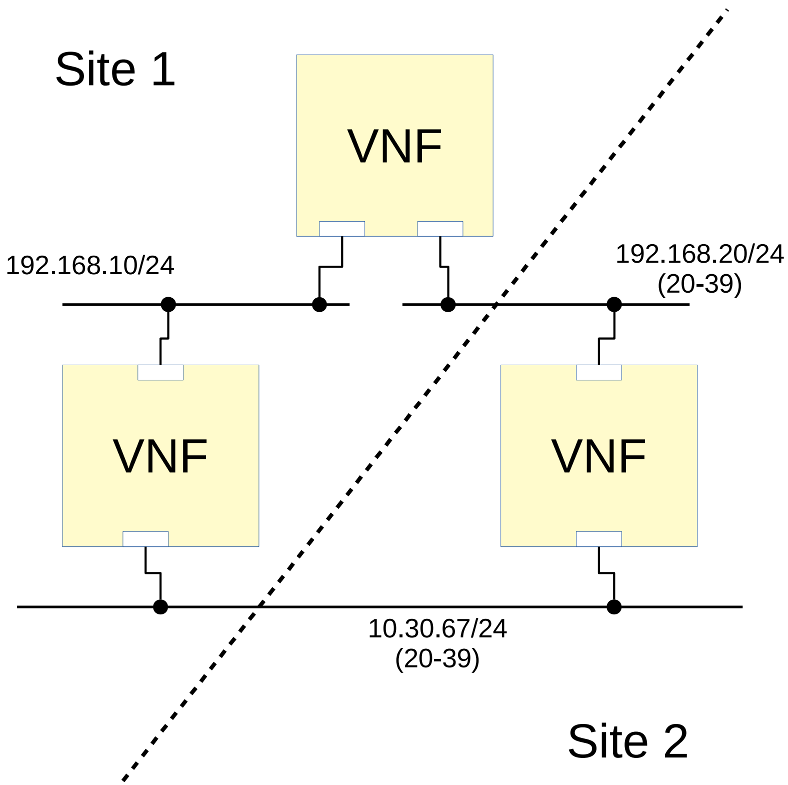
- A VNF could consist of multiple and variable VDUs (scaling)
- VL(D) profiles:
  - Subnet (e.g., 192.168.10/24)
  - DHCP range (e.g., 30-40)
  - Some defined by VNFD/NSD providers
  - Rest defined at deployment

Multi-VIM IP pool split



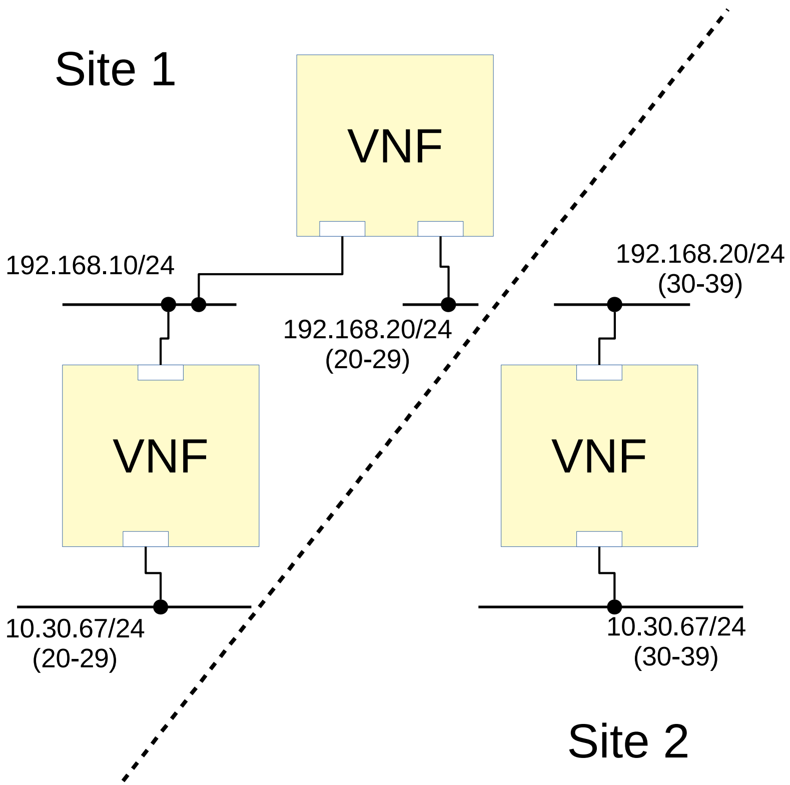
- Express as NSD
- Deploy it
  - Assign VNFs to different VIMs
- OSM 5/6 implementation
  - Leads to WIM interaction
  - No IP address co-ordination

Multi-VIM IP pool split



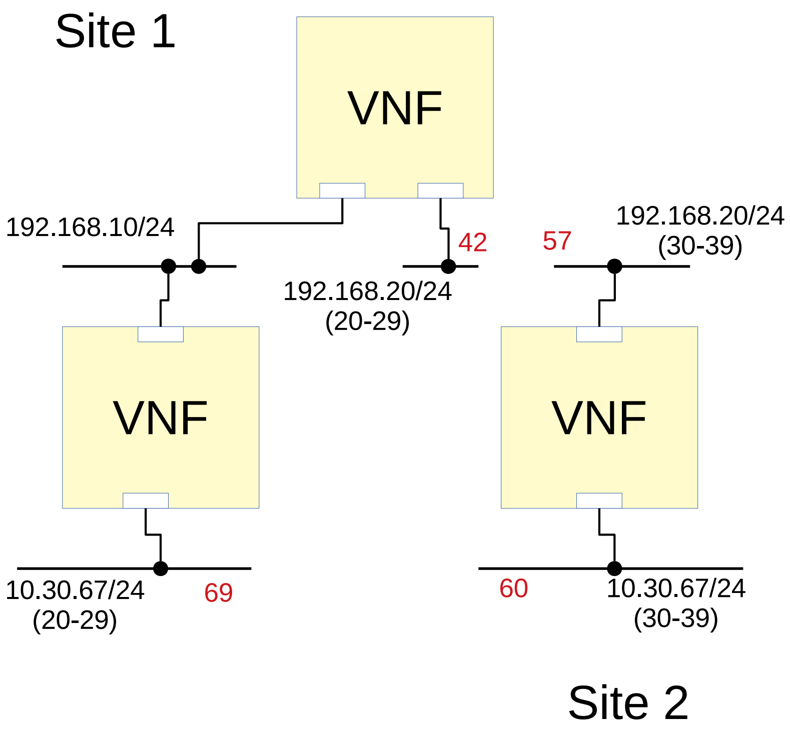
- No VNF spans two or more sites
- No internal VL spans sites
- Some external VLs span sites
  - Some may span more than two
  - A split VL will need representation at each site
- VL profiles must be defined before splitting
  - Representations of the same VL at different sites must be compatible
  - Representations of different VLs at different sites must be distinct
  - To permit L2 inter-site connectivity

Multi-VIM IP pool split



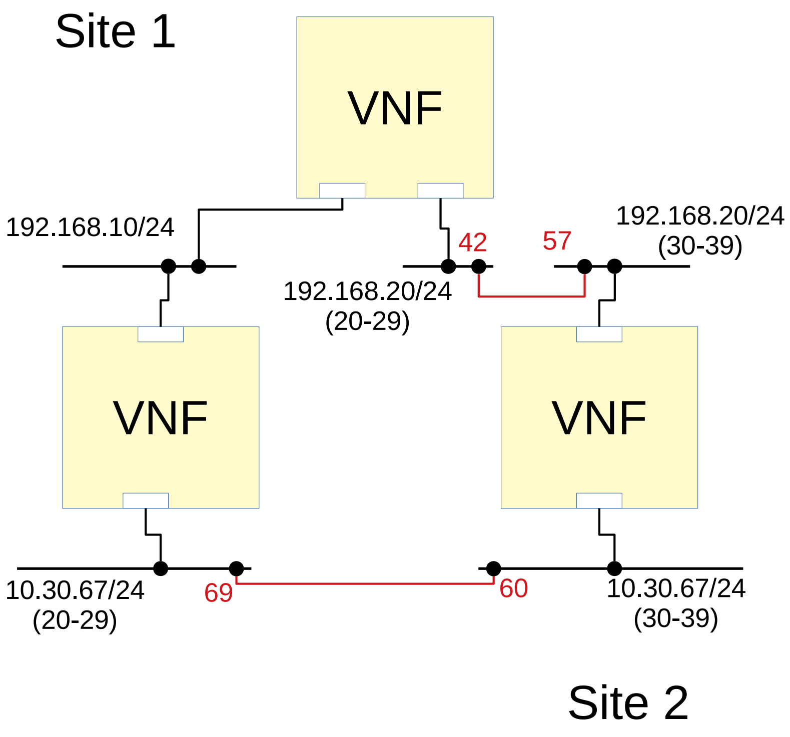
- Each OpenStack site provides a DHCP agent for each VL it represents
  - One address is used as the default gateway, DNS server and DHCP server
  - Agent only responds to DHCP requests of MACs known locally to use that network
  - No awareness of DHCP at other site
  - DHCP ranges for same VL at each site must not overlap!
  - DHCP ranges must anticipate scaling

Multi-VIM IP pool split



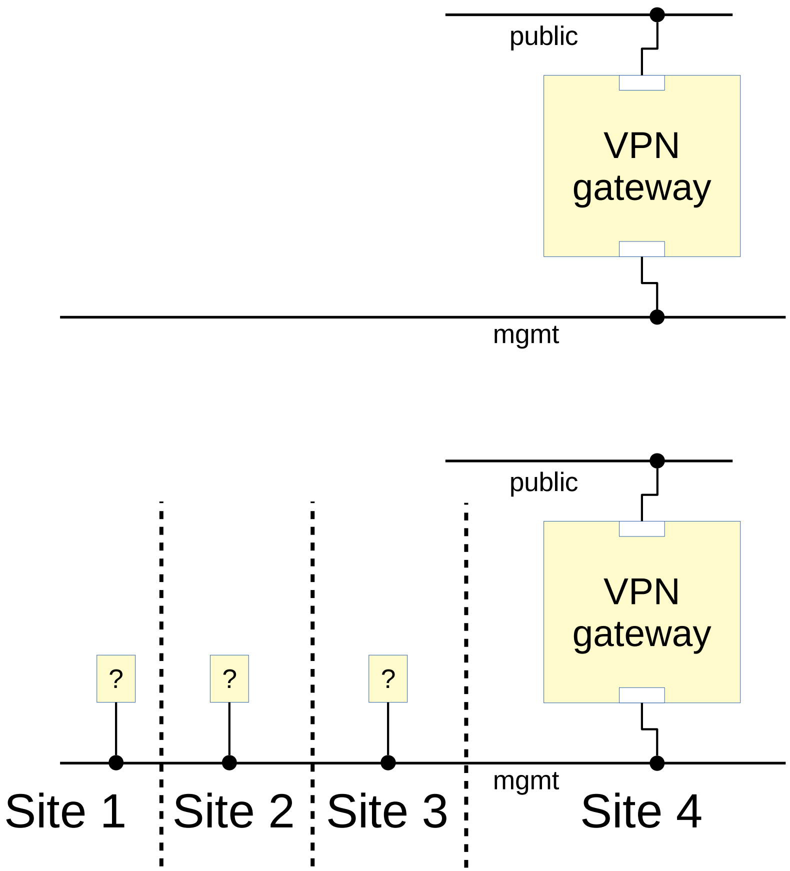
- Inter-site connectivity
  - Get VLAN tags of VIM representations of multi-site VLs
    - 42 & 57
    - 69 & 60
  - Add site identification as context
    - Site 1.42 & Site 2.57
    - Site 1.69 & Site 2.60
  - Estimate bandwidth at each end point
    - Site 1.42 (10M) & Site 2.57 (10M)
    - Site 1.69 (10M) & Site 2.60 (10M)
  - Supply to WIM

Multi-VIM IP pool split



- Site 1.42 (10M) & Site 2.57 (10M)
- Site 1.69 (10M) & Site 2.60 (10M)
- Broadcasts are visible across both sites
  - ARPs work
  - DHCP requests seen by both agents, but only one responds

New management networks through OSM



- Define a VLD
  - Include a VPN gateway as a VNF
- Deploy across sites
  - But only VNFs can be assigned to VIMs
- Create tenant-specific VIMs using new network as default management