Cloud networking (VITMMA02)
DC network topology, Ethernet extensions

Markosz Maliosz PhD

Department of Telecommunications and Media Informatics
Faculty of Electrical Engineering and Informatics
Budapest University of Technology and Economics

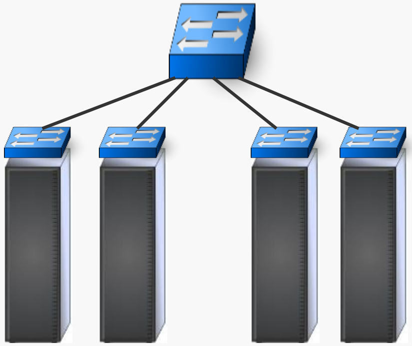
Spring 2020

Data Center Traffic Patterns

- Traffic flow
  - north-south: between servers and core switch
  - east-west: between servers
    - e.g. VM migration, storage replication
- Request-response communication
  - before: a client request is responded by a single server
  - today: a client request is responded by many interactions of servers
    - e.g. a Google map search request
      - send information to a local search engine
      - based on the result, gather appropriate map data from map server
      - search, retrieve and display relative nearby places
      - retrieve related information about the client based on recent web transactions
      - send targeted advertisement

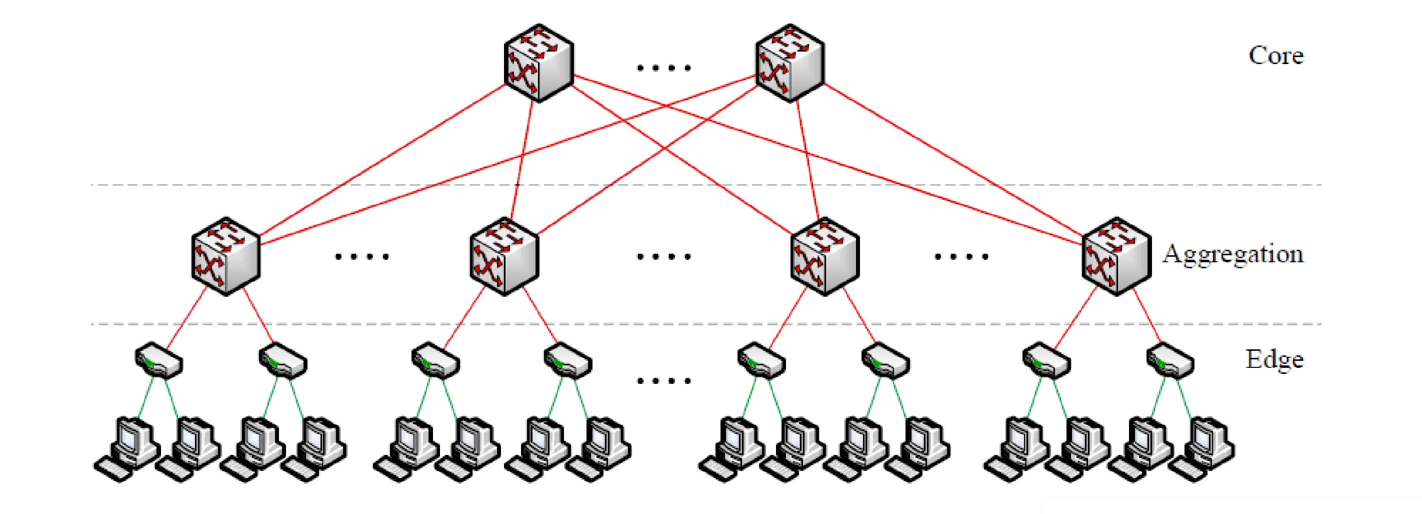
Network Topology

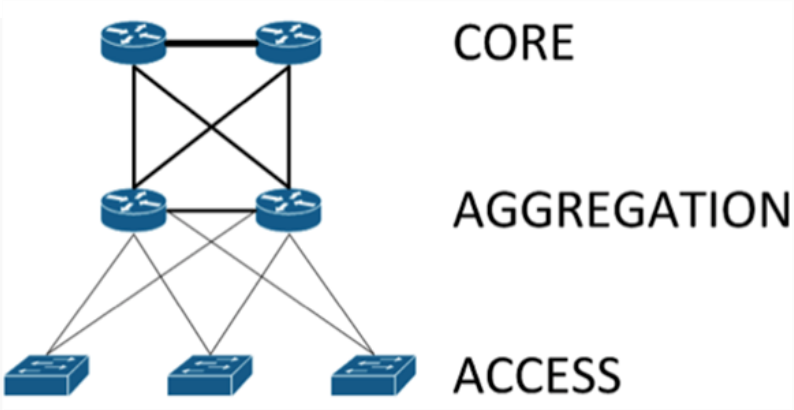
- 3 level hierarchy: ToR, aggregation, core switch
- flat (ter) topology, 2 levels: ToR and core switch
  - single large core switch: expensive, limited number of ports
    - e.g. price of a 128 port GbE switch is approx. 100-times of a 48 port switch



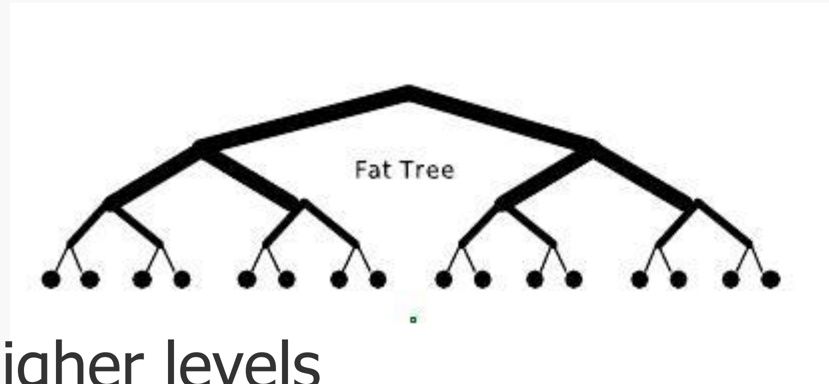
Network Topology

- Redundancy and/or load balancing
  - dual star

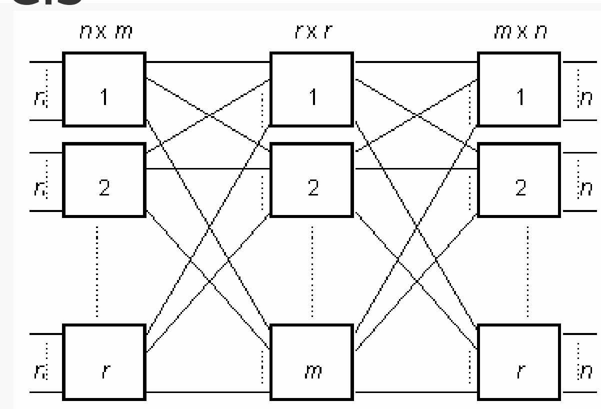




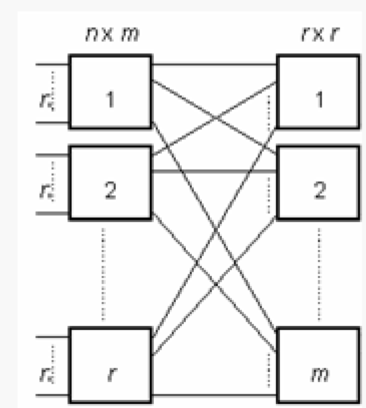
Fat-tree topology



- Fat-tree
  - 1:1 oversubscription
  - bandwidth is added up on higher levels
  - different port numbers
  - multistage switching
  - Charles Clos 1952, for telephone switching system



- Folded multistage switching
  - folded Clos
  - merged input and output
  - also called fat-tree



Fat-tree topology in the data center

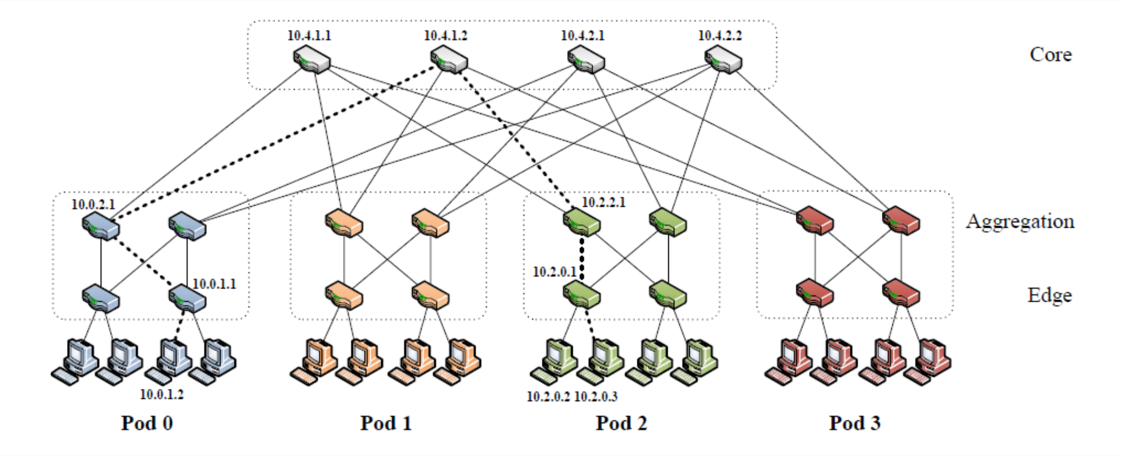
- full mesh: complex cabling
- leaf and spine switches
- load balancing by spine switches, ECMP
- can be built by identical switches with N ports
  - leaf ports: N/2 downstream, N/2 upstream (max. N/2 spine switches) – 1:1 oversubscription
    - that's why it is called fat-tree
  - spine: N ports ⇨ max. N leaf switches
  - altogether up to
    - 1.5xN switches
    - NxN/2 servers connected to leaf switches

Fat-tree topology in the data center

- Load balancing
  - ideal case: traffic is distributed uniformly on spine switches
  - reality
    - flow based load balancing
      - round robin
      - hash
    - jumbo frames (9kB)
    - leaf switches are uncoordinated
- Resiliency
  - spine switch failure
    - all connections are up but with reduced bandwidth
  - leaf switch failure
    - connected servers are unavailable
    - protection: multi-homing = dual NIC, each connected to different leaf switch

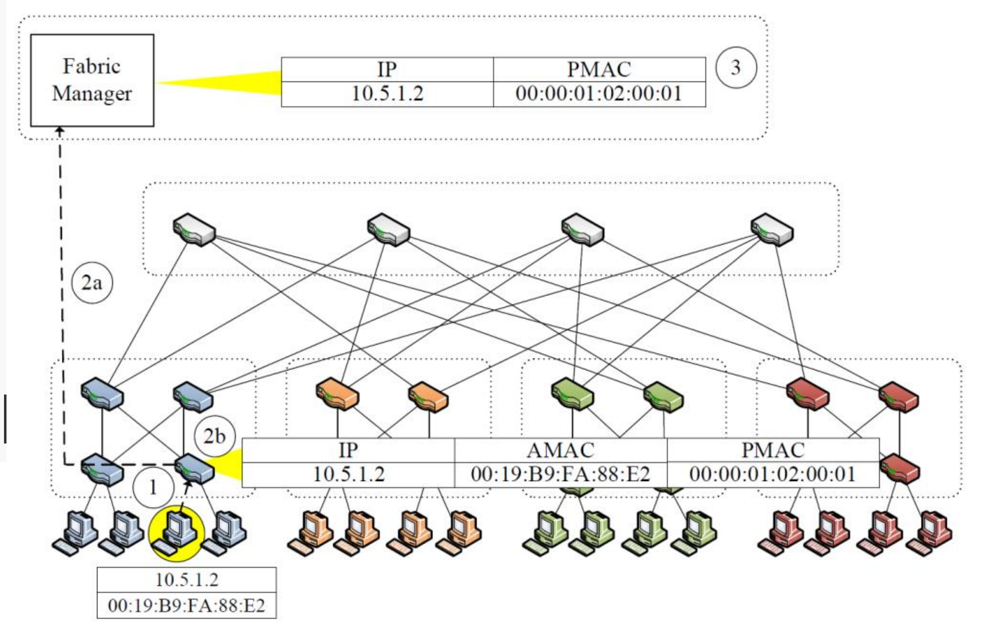
Fat-tree topology in the data center

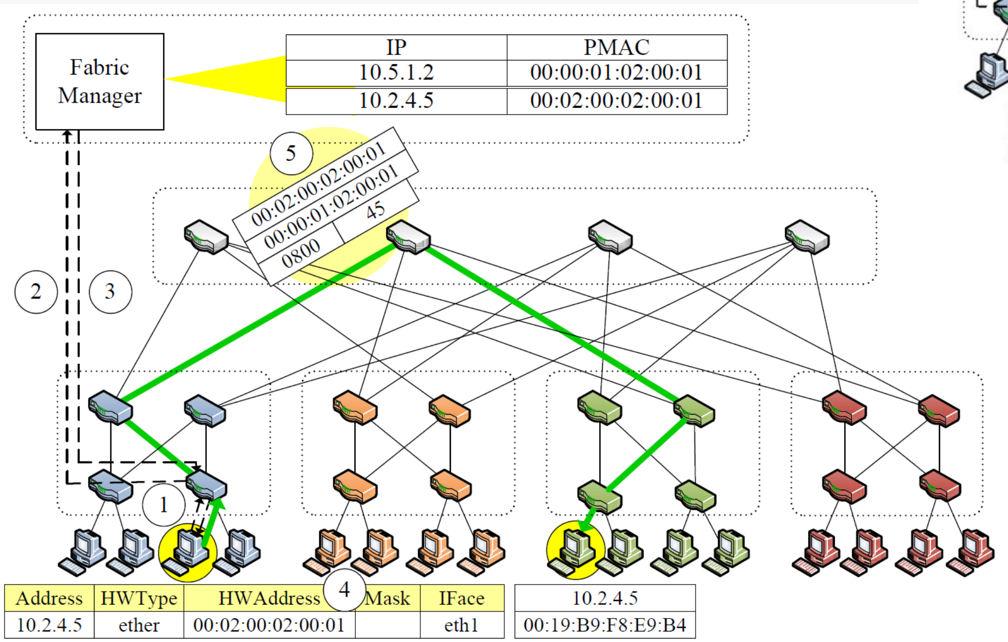
- A topology scheme
  - switches with k ports
  - k pod (group)
  - k/2 edge and aggr. switch / pod
  - core switches connected to each pod
    - in k/2 units via aggr. switches
  - $k * k/2 * k/2 = k^3/4$ servers
  - $k*k + (k/2)^2 = 5/4\ k^2$ switches
  - $(k/2)^2$ ECMP path
  - figure: k=4
  - k=48
    - 27 648 servers
    - 2 880 switches
    - 576 ECMP path



Addressing based on L2 topology

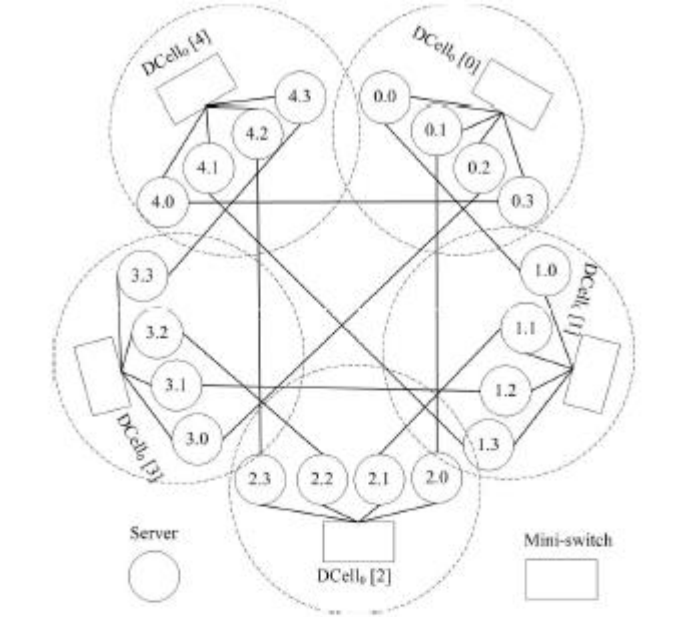
- Portland
- Pseudo MAC (PMAC)
  - topology based:
    - pod:position:port:vmid
- Fabric manager
  - handling ARP requests
- Location Discovery Protocol





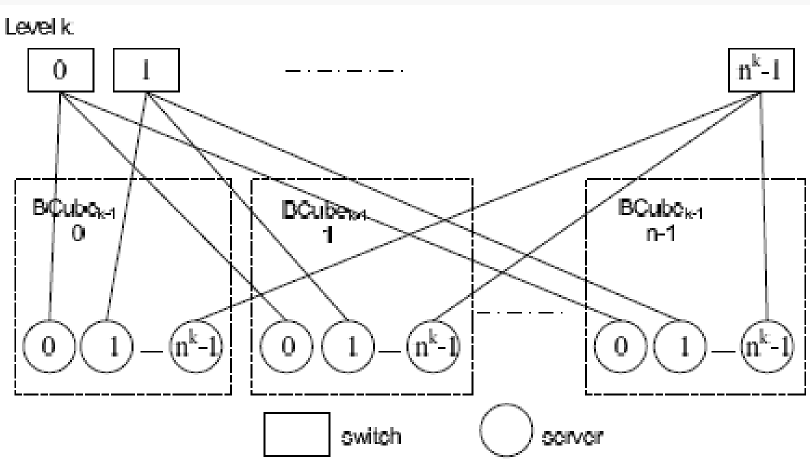
Hybrid networks: servers and switches

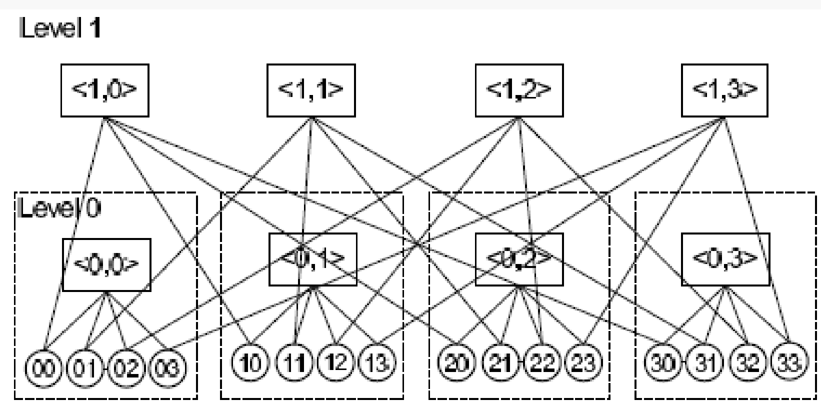
- Recursive topology model: DCell
- Incremental expansion
- Levels
  - 0. level: **n** server and **1** switch
  - k+1. level: (# of k. level servers +1) level k cells connected in full mesh
- Hybrid networking
  - intra-cell: via switch
  - inter-cell: servers are used as routers
    - at first the route between the same level cells containing the source and destination is determined, then the intra-cell route
    - not a min hop routing
- Robust
  - many alternative routes
- Performance
  - bandwidth depends on the size of the network
  - more intermediate hops



Hybrid networks: servers and switches

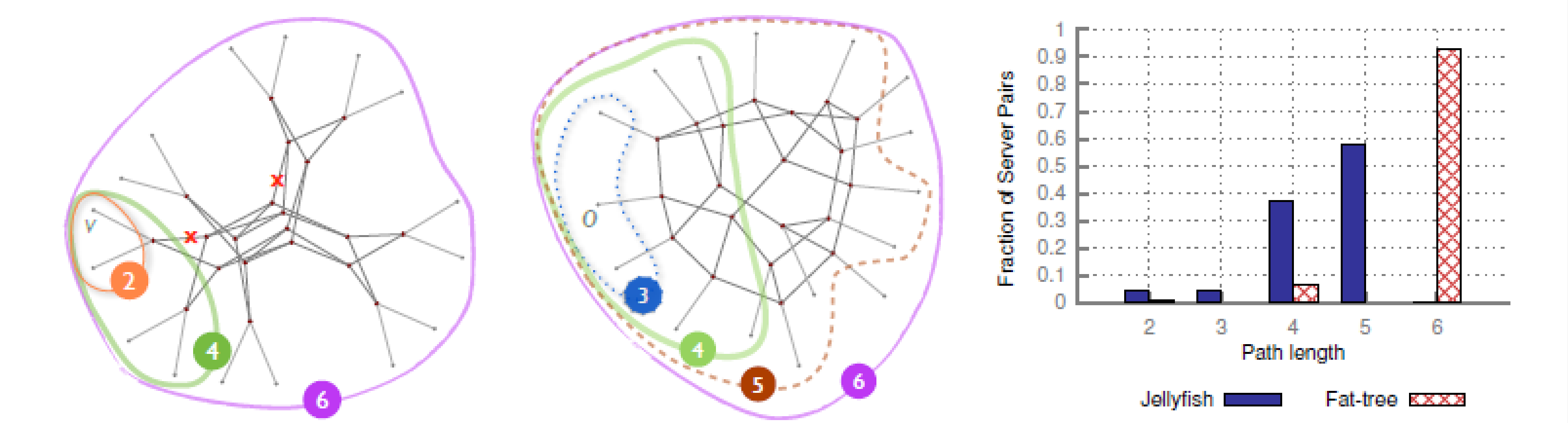
- BCube: for modular data center units installed into containers
  - number of servers in the order of 1000s
- Properties
  - graceful degradation in case of failure
  - small diameter network
  - a lot of parallel connections between servers
  - source routing
    - multipath
    - network probes
- Recursive topology model
  - Levels
    - 0.: **n** servers interconnected by a n port switch
    - k.: **n** k-1. level BCube and $\mathbf{n^k}$ n port switch
  - k. level
    - $\mathbf{n^{k+1}}$ server
    - servers: k+1 port
    - k+1 level from switches, $\mathbf{n^k}$ n port switch at each level





Jellyfish topology

- ToR switches connected by a random graph
- Incremental expansion
- Switches with different port numbers
- Advantages
  - average path length is smaller
  - with the same number of switches more servers are connected compared to fat-tree topology



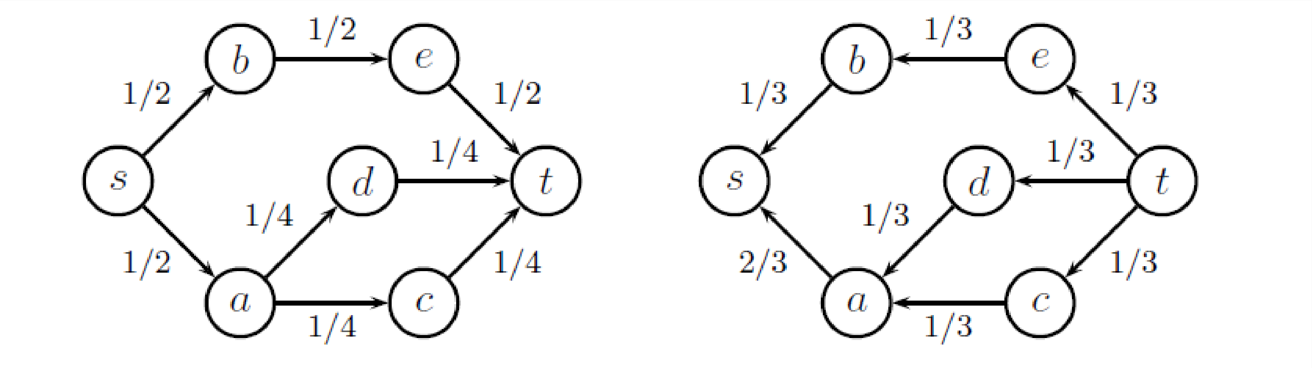
(a) (b) (c)

Improving network utilization

- Ethernet Spanning Tree Protocol
  - spanning tree: unused links
  - Rapid STP (RSTP)
  - Multiple STP (MSTP)
  - ideal for arbitrary and changing topologies
- But not ideal for data centers
  - structured and not frequently changing
  - new standards
    - Equal Cost MultiPath (ECMP) routing
    - Shortest Path Bridging (SPB)
    - Transparent Interconnection of Lots of Links (TRILL)

ECMP

- Equal Cost MultiPath

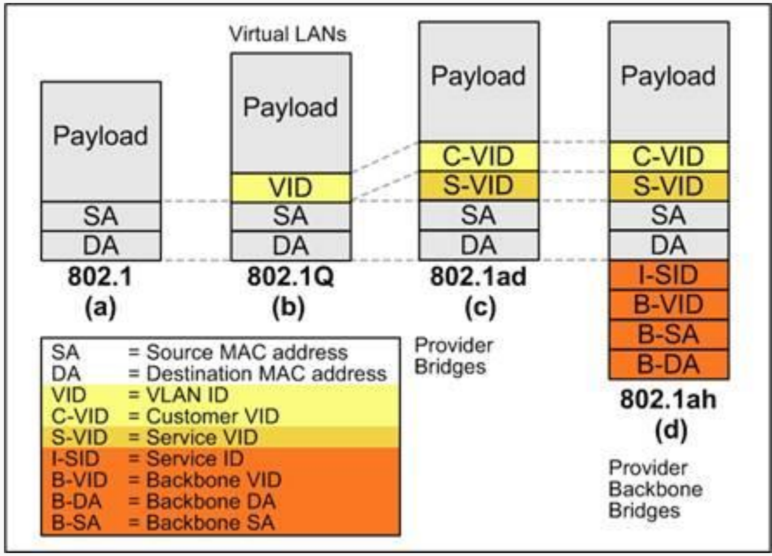


- Layer3 routing or tunneling between Layer2 domains
  - L2 over L3
- generally not used in networks
  - if routes join before the destination, only the complexity is enlarged, but not the bandwidth utilization
  - virtual network ⇔ physical network

Shortest Path Bridging

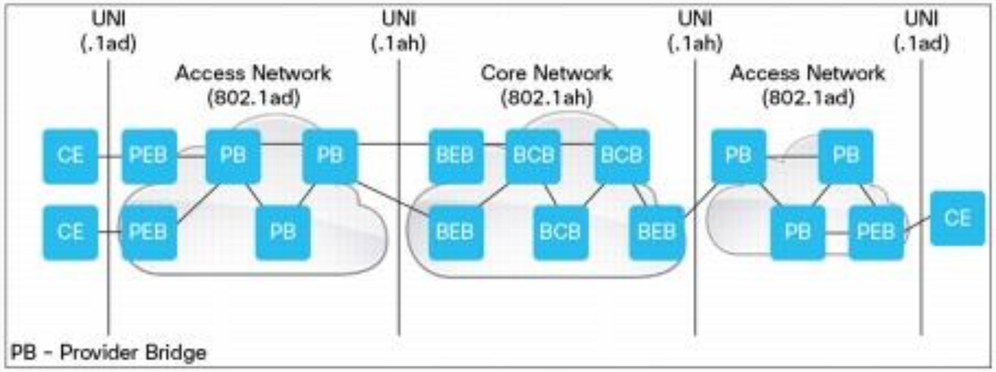
- Origins: Carrier Ethernet
  - Provider Bridging (PB) 802.1ad
  - Provider Backbobe Bridging (PBB) 802.1ah
- Shortest Path Bridging (SPB) 802.1aq

Carrier Ethernet



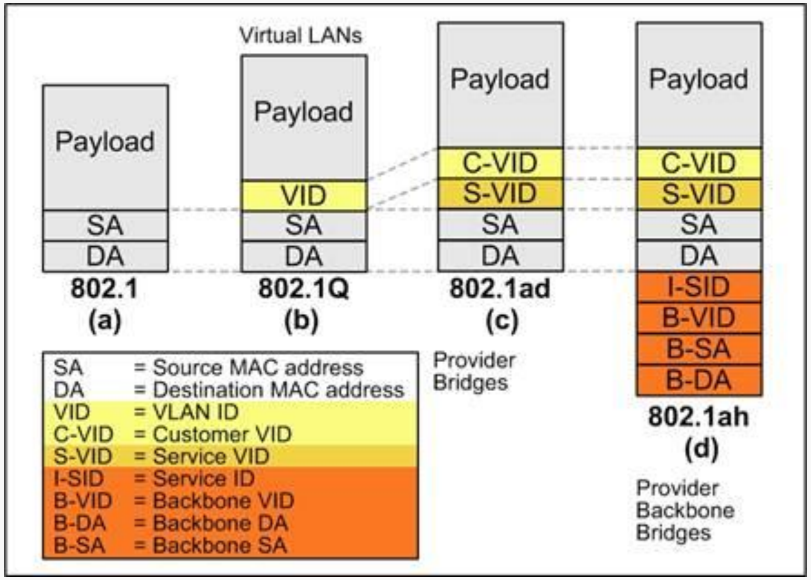
Ethernet in carrier networks (MAN, WAN)

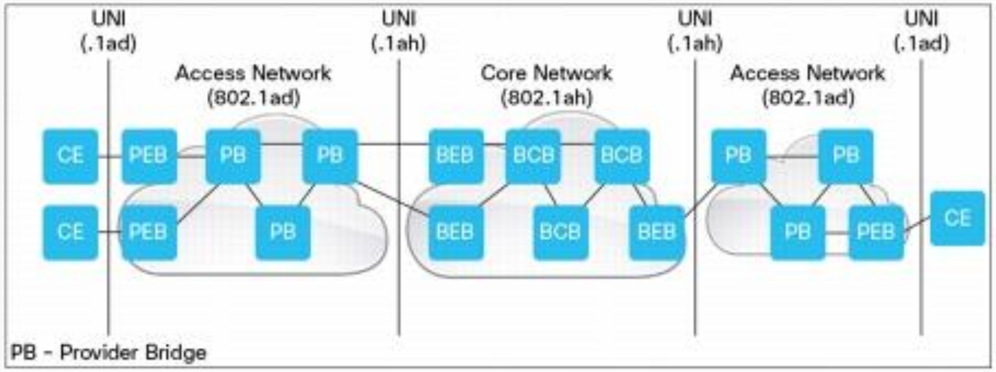
- Ethernet service for many customers
  - separating customers
- tunneling by additional tags
  - keeping customer VLAN information
  - separating service instances (customers) (PB)
    - Q-in-Q: Customer tag, Service tag
    - two VLAN IDs (VID)
    - 4096 service instance (upper bound)
  - complete separation of customer and provider domains (PBB)
    - MAC-in-MAC: separated address space
      - customer addresses are not seen by switches in the carrier network
    - service tag: 24 bit I-SID (service identifier) 16M service instances
    - separating service and transport layers: I-SID and B-VID



Carrier Ethernet

- Mapping virtual networks at the edge
  - C-VID ⇨ S-VID ⇨ I-SID ⇨ B-VID
  - Edge Brigdes
- In the core network: forwarding based on VLAN ID and destination MAC address
  - Core Bridges

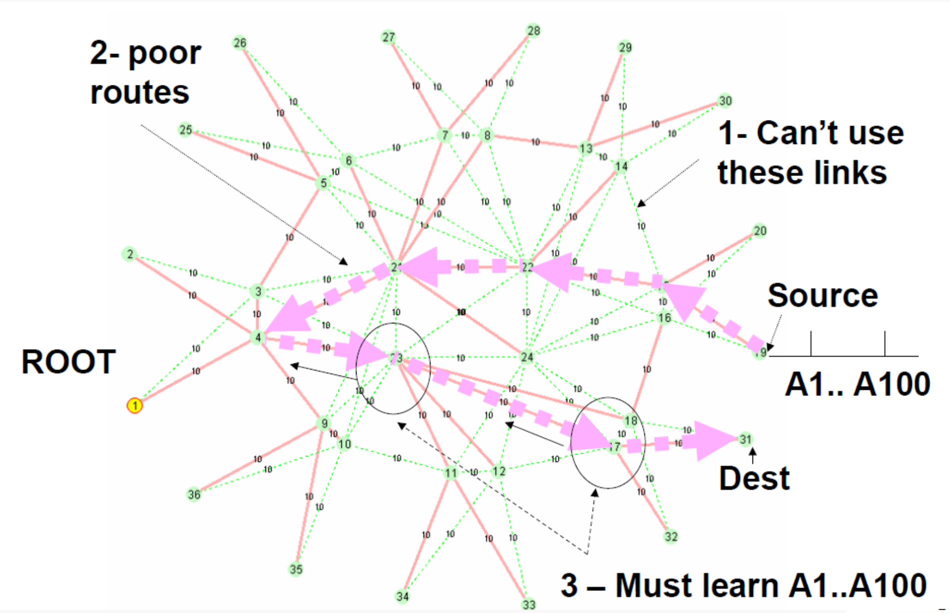


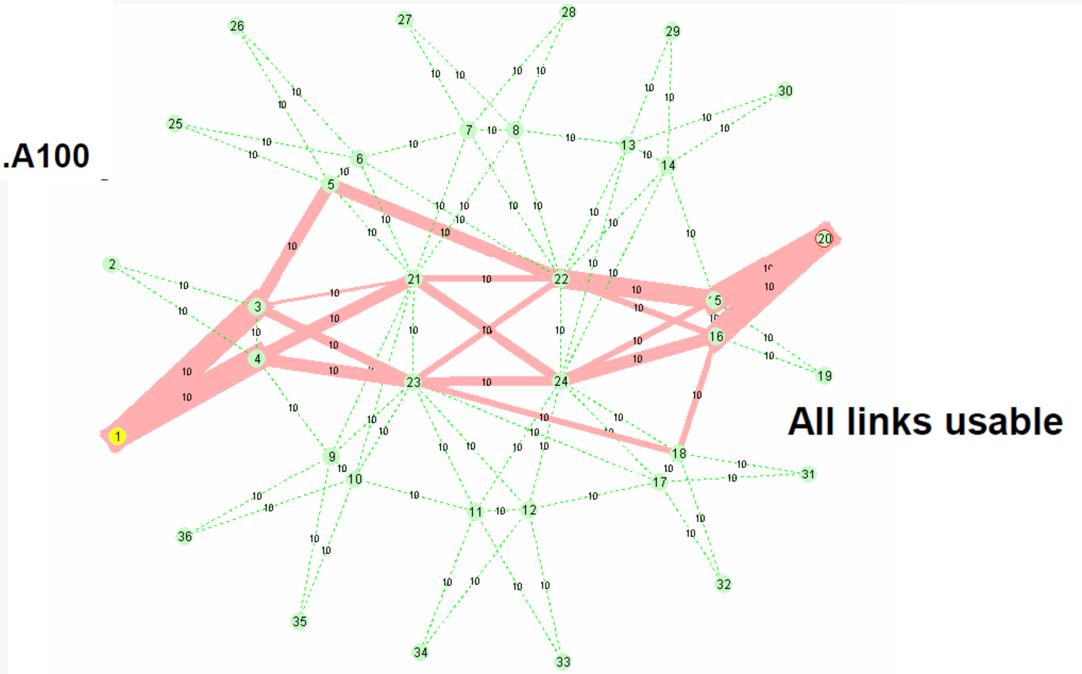


Shortest Path Bridging

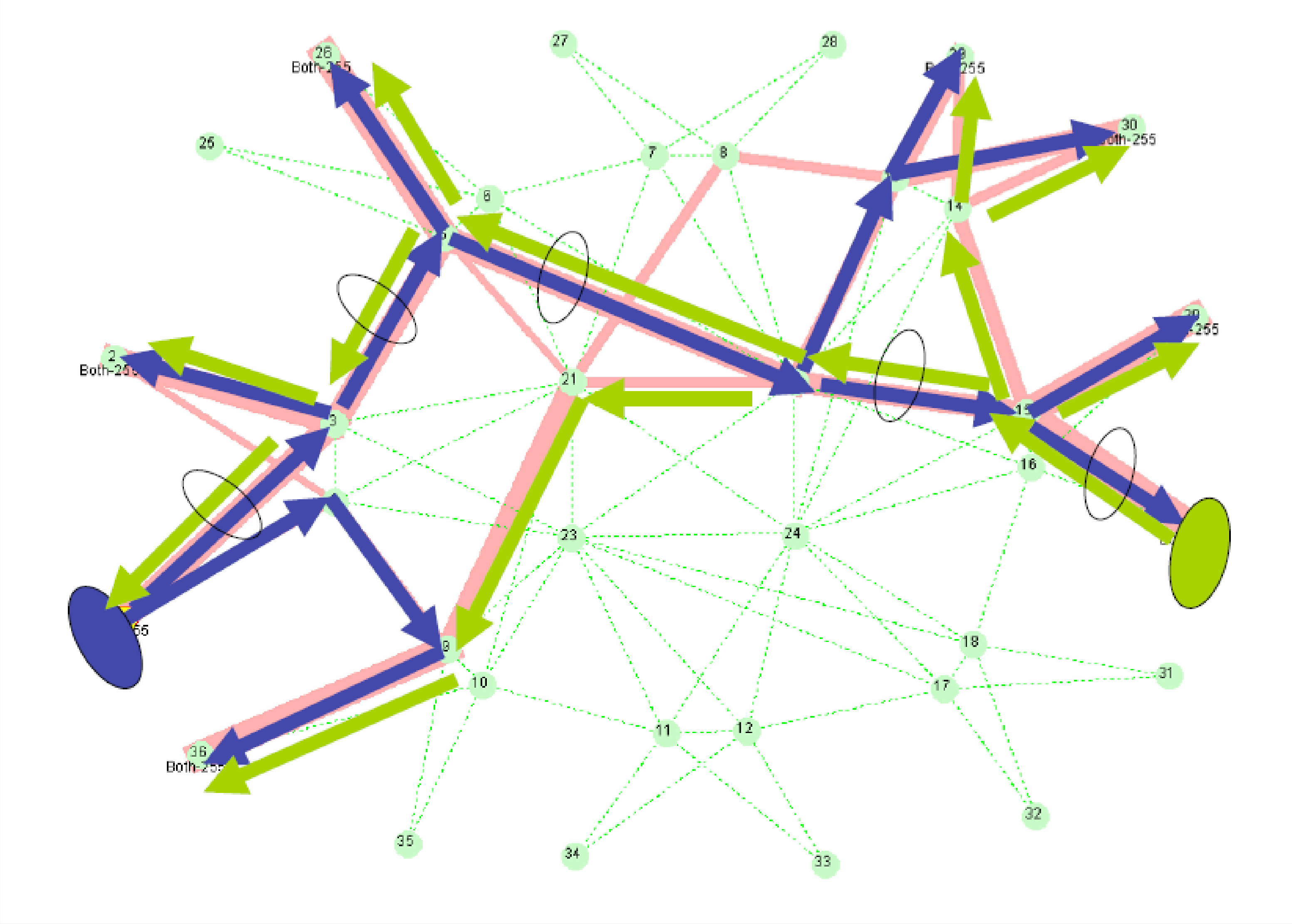
- Replacing STP with a new control plane
  - providing logical networks over native Ethernet
  - link state protocol advertising the topology and the logical network membership
    - Intermediate System to Intermediate System (IS-IS) with extensions with link state protocol
      - runs directly at Layer 2
      - no IP addresses are needed, as they are for OSPF
      - IS-IS can run with zero configuration
      - with TLV (type, length, value) encoding new types of data
    - automatic link state discovery
    - no blocked ports, links
  - using equal cost *multiple* shortest paths
  - sources calculate a shortest path tree
    - symmetric forward-backward paths
- Encapsulation
  - PB ⇨ SPB Vlan ID (SPBV)
  - PBB ⇨ SPB MAC (SPBM)
- vendors: Avaya, Huawei, Alcatel-Lucent

STP vs. SPB


2- poor
routes
1- Can't use
these links
ROOT
Source
A1.. A100
Dest
3 – Must learn A1..A100


All links usable

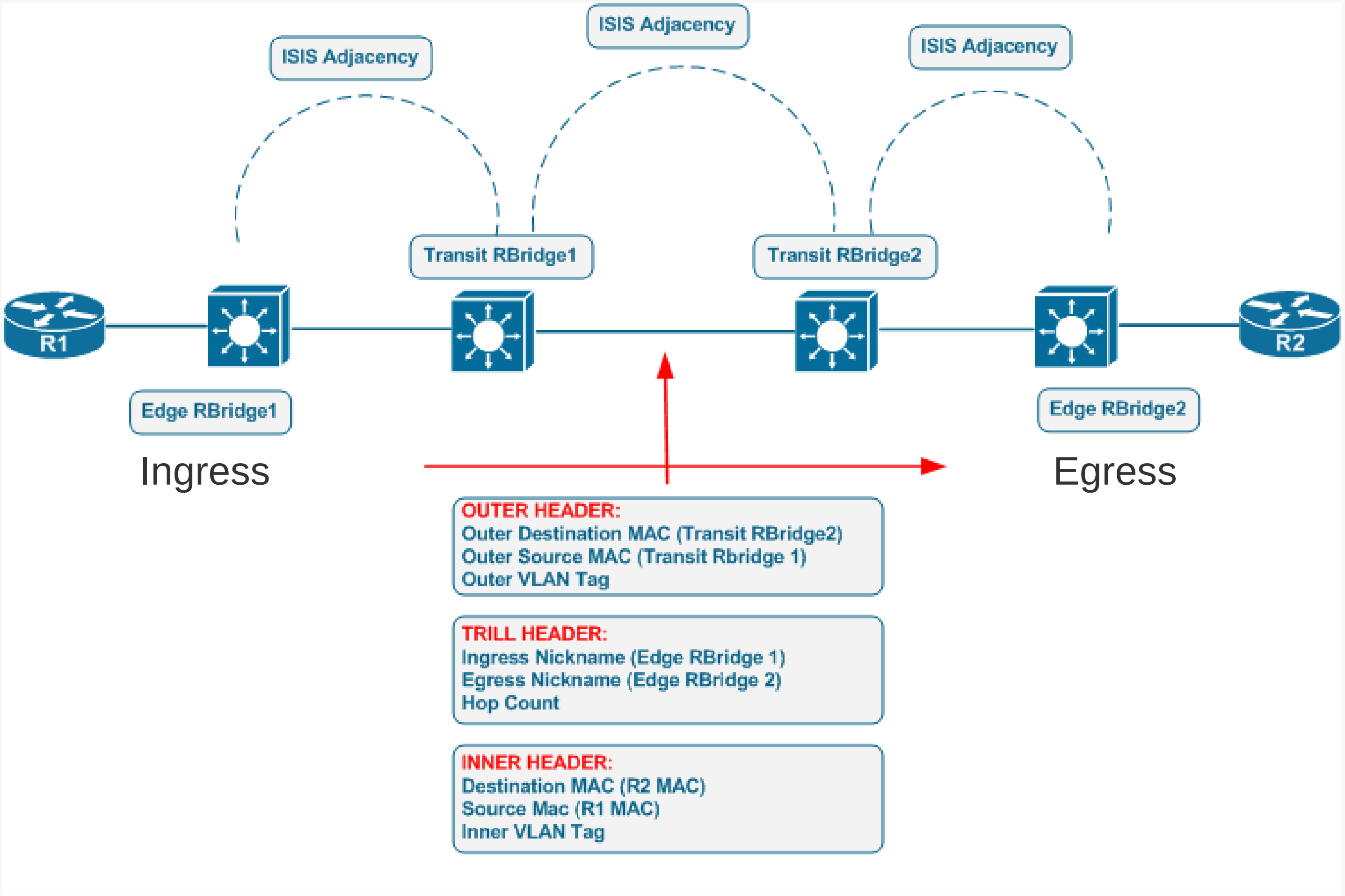
SPB


26
Both-255
27
28
255
30
oth-255
25
7
8
6
14
255
2
Both-25
21
3
15
16
23
24
55
9
10
18
31
17
11
12
36
Both-25
35
34
33
32

TRILL

- Transparent interconnection of lots of links
  - RBridge: routing bridge
    - *multipath* (ECMP) tunnels over L2 domain
  - with link state protocol: IS-IS
    - for same reasons as in SPB
  - extra headers
    - TRILL header
      - hop count
      - RBridge ingress, egress nickname
    - outer Ethernet header
      - RBridge source, destination MAC
      - VLAN tag
  - transit/relay RBridges swap the outer Ethernet header to the next hop RBridge MAC address
    - standard Ethernet switches forward traffic by outer MAC address
- vendors: Cisco, Brocade

TRILL


ISIS Adjacency
ISIS Adjacency
ISIS Adjacency
Transit RBridge1
Transit RBridge2
R1
R2
Edge RBridge1
Edge RBridge2
Ingress
Egress
OUTER HEADER:
Outer Destination MAC (Transit RBridge2)
Outer Source MAC (Transit Rbridge 1)
Outer VLAN Tag
TRILL HEADER:
Ingress Nickname (Edge RBridge 1)
Egress Nickname (Edge RBridge 2)
Hop Count
INNER HEADER:
Destination MAC (R2 MAC)
Source Mac (R1 MAC)
Inner VLAN Tag

SPB vs. TRILL

	SPB	TRILL
Standardization	IEEE	IETF
Data forwarding	Ethernet switching Without MAC address swapping	Forwarding by RBridge nicknames MAC address swapping hop-by-hop
Virtual networks	SPBM: 16 million	4096, with optional header: 16 million
Hardware	Existing, low cost Ethernet ASIC	New hardware
Loop protection	Reverse Path Forward Checking (RPFC)	RPFC + hop count
ECMP	Yes, 16 way	Yes, 16 way

Reverse Path Forward Checking

- checking whether a source addresses can be reached via the input interface (is there an entry in the forwarding table in the opposite direction), i.e. it arrived on the shortest path
  - if yes: forward
  - if not: drop
- conditions
  - correct forwarding information in a converged state
  - symmetric forward-backward paths
- unicast and multicast

Sources

- Shortest Path Bridging, IEEE 802.1aq, Tutorial and Demo, NANOG 50 Oct 2010, Peter Ashwood-Smith, Huawei
- Radhika Niranjan Mysore, Andreas Pamboris, Nathan Farrington, Nelson Huang, Pardis Miri, Sivasankar Radhakrishnan, Vikram Subramanya, and Amin Vahdat. PortLand: a scalable fault-tolerant layer 2 data center network fabric. *SIGCOMM Comput. Commun. Rev.* 39, 4 (August 2009)
- Ankit Singla, Chi-Yao Hong, Lucian Popa, and P. Brighten Godfrey. 2012. Jellyfish: networking data centers randomly. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation* (NSDI'12). USENIX Association, Berkeley, CA, USA.