

Address Assignment for Stateless Flow-Zone Switching in the Data Center

omniran-18-0059-00-CQ00

Document Number:

Date Submitted:

2018-07-10 (originally 2018-01-24)

Source:

Roger B. Marks

EthAirNet Associates

404 Montview Blvd

Denver, CO 80207 USA

*<http://standards.ieee.org/faqs/affiliationFAQ.html>>

Voice:

+1 802 227 2253

E-mail:

roger@ethair.net

Re: 802.1CQ Project, 802.1 Data Center Bridging Task Group

Venue:

July 2018 Plenary Session

Abstract:

This document describes a local address assignment algorithm suitable for use in a data center environment. The application of the algorithm supports stateless Layer 2 routing, without the need for stored forwarding tables. This contribution is intended for discussion regarding its suitability as the basis of a proposal for a standardized address assignment method for specification in P802.1CQ (“Draft Standard for Local and Metropolitan Area Networks: Multicast and Local Address Assignment”).

Notice:

This document represents the views of the author and is offered as a basis for discussion.

Address Assignment for Stateless Flow-Zone Switching in the Data Center

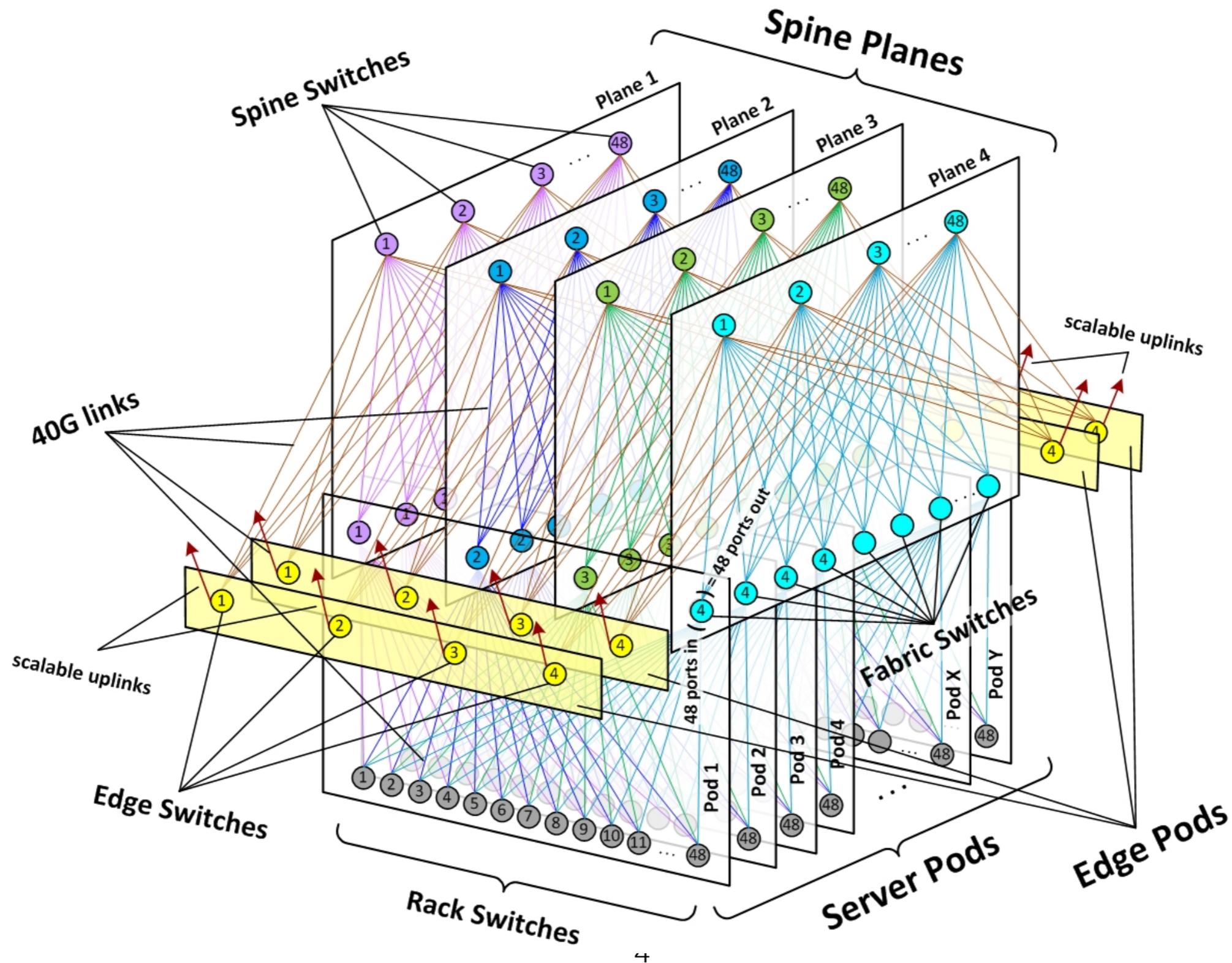
Roger B. Marks
EthAirNet Associates

Key points

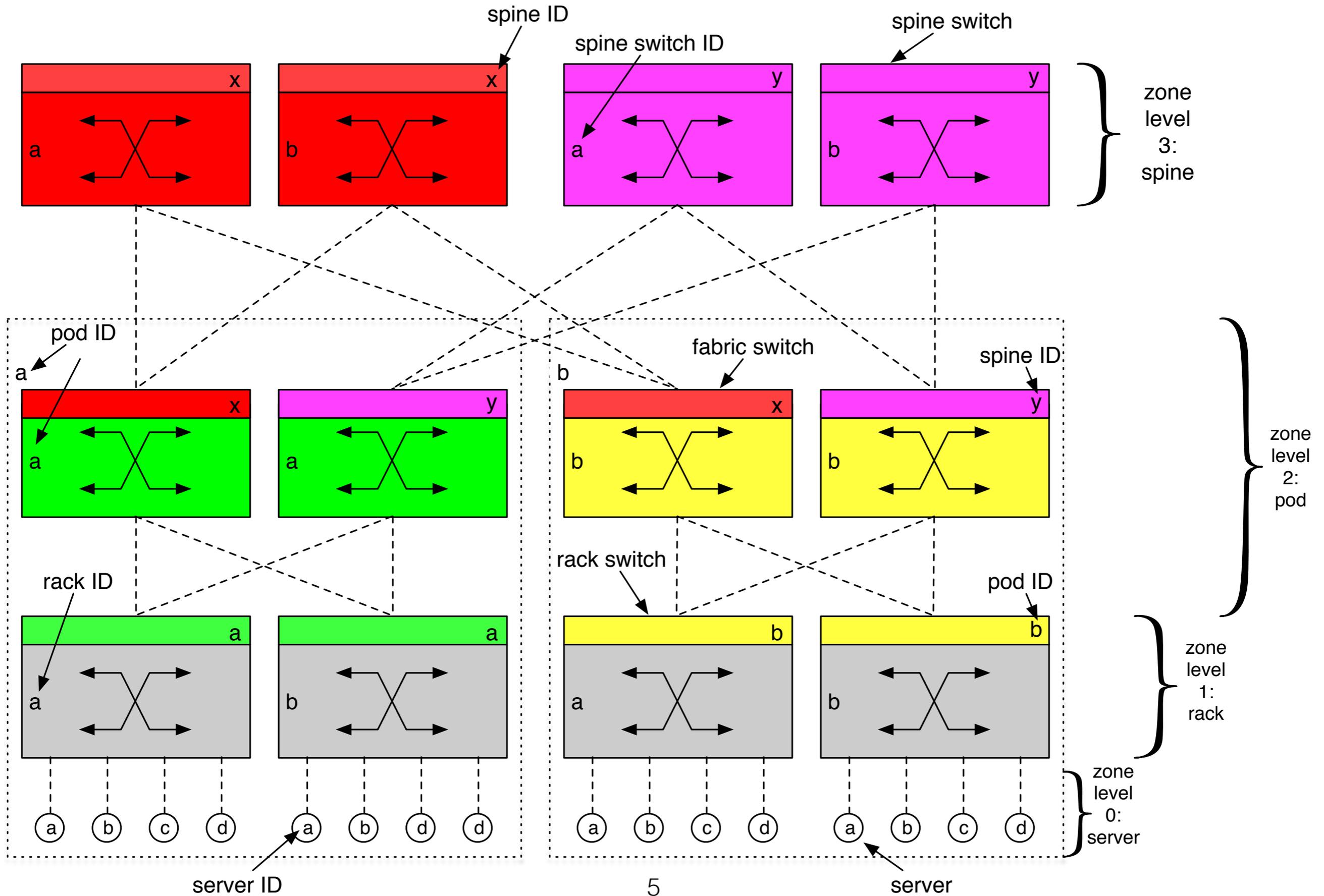
- A Layer 2 data center routing method is described at a high level.
- The method minimizes state in switches by embedding routing instructions in MAC addresses.
- A specific version can eliminate the need for forwarding tables.
- The method relies on specific address assignment, which could be considered for standardization in P802.1CQ.

Data Center folded-Clos “fat-tree” architecture

(example: Layer 3 [Facebook Data Center Fabric](#))



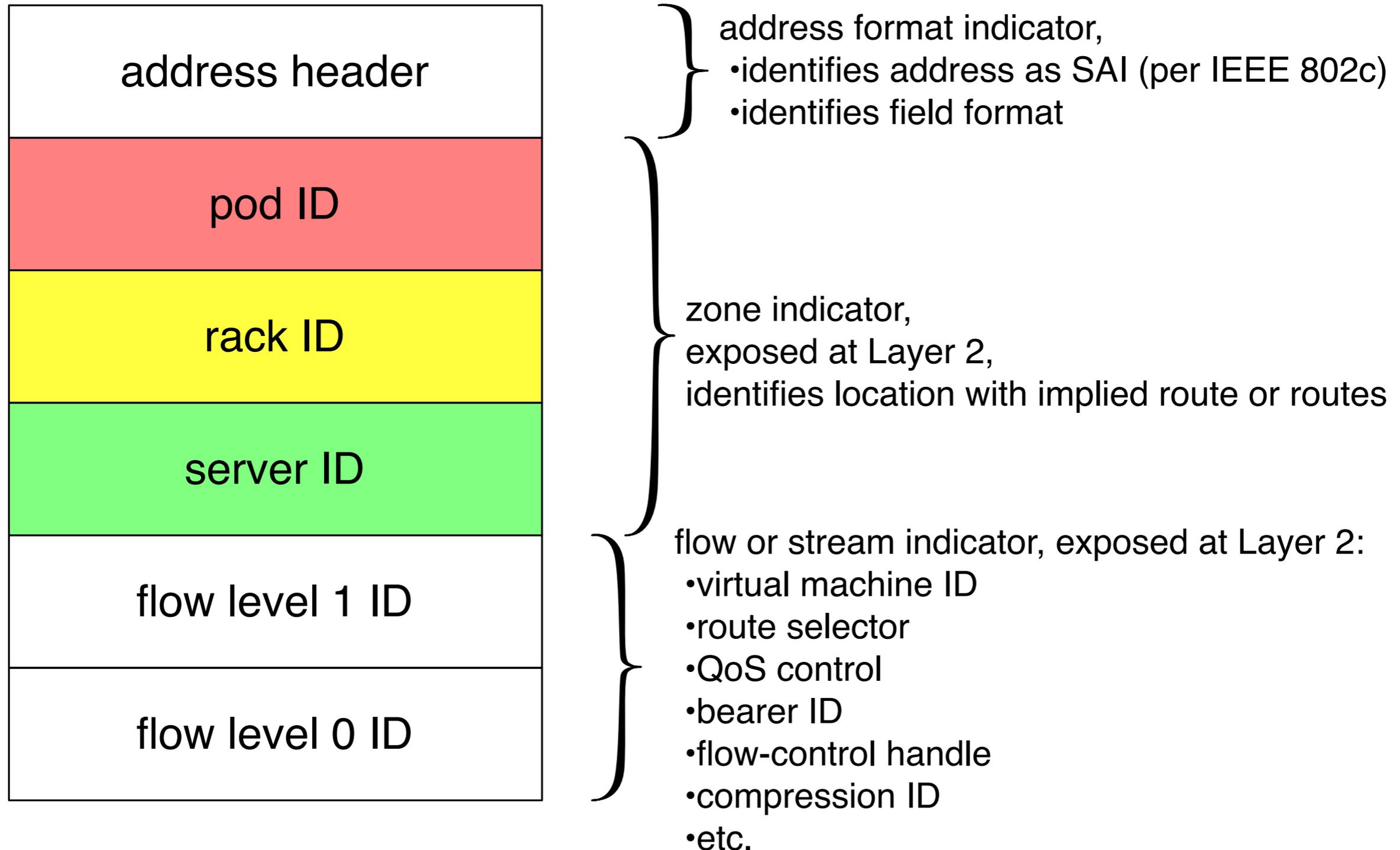
Data Center architecture: Four Zone Levels



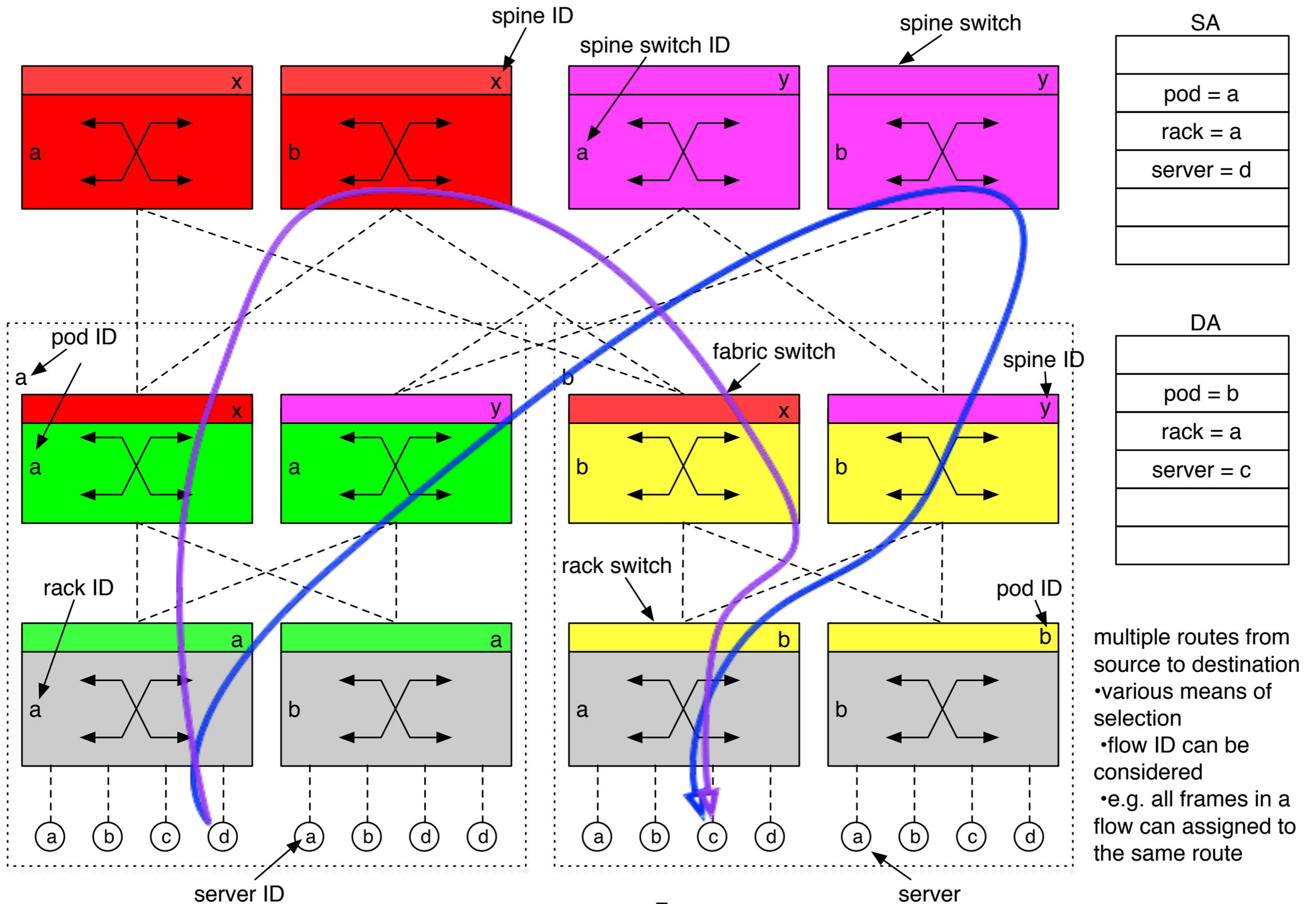
Flow-Zone Data-Frame Address Format

e.g., for 48-bit data-plane frames addressed to servers

server address



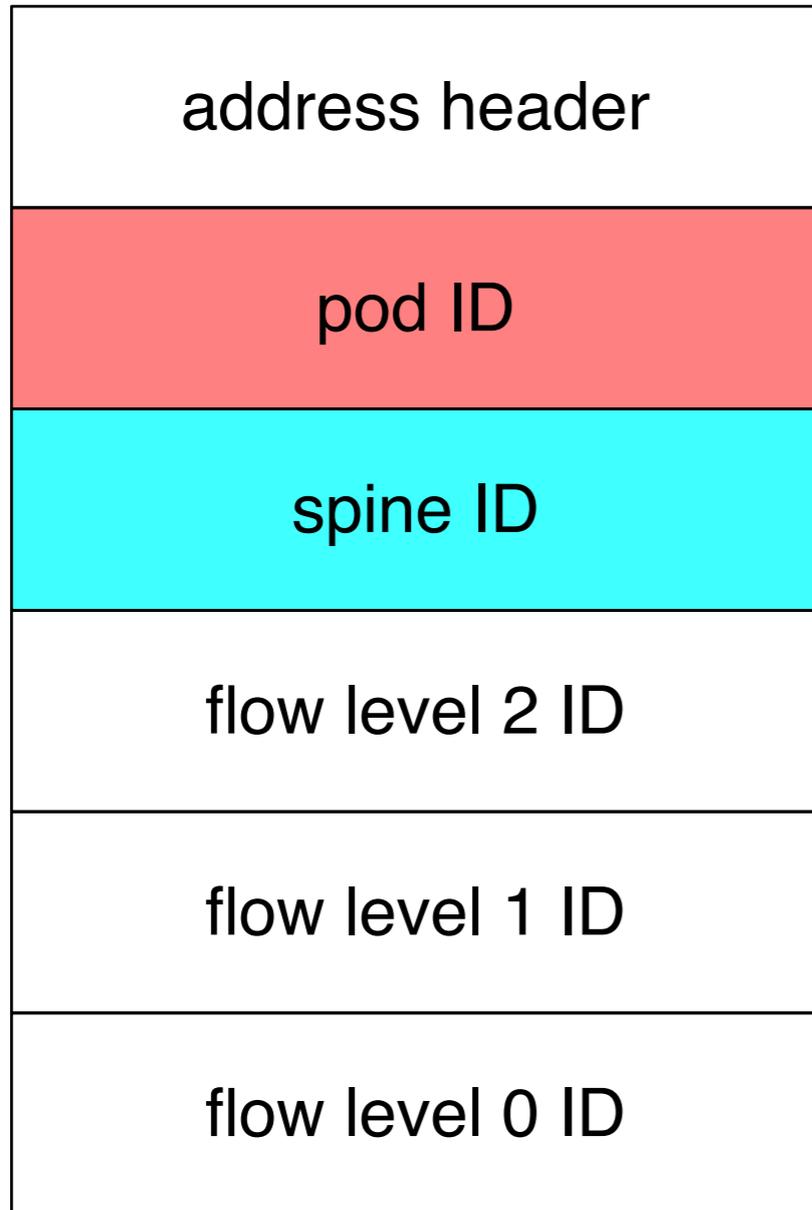
Flow-Zone Switching: DA-embedded Source Routing



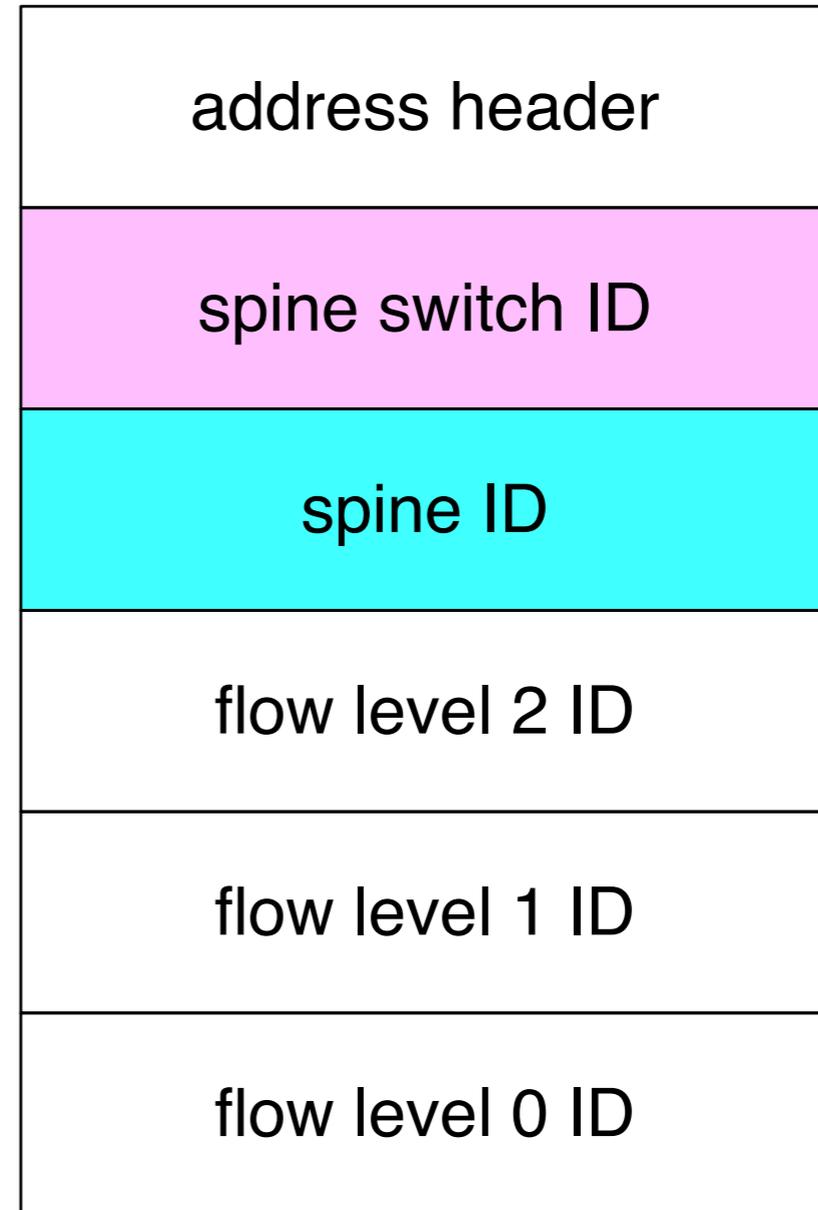
Flow-Zone Control-Frame Address Format

for control-plane frames: addressed to switches

fabric switch address



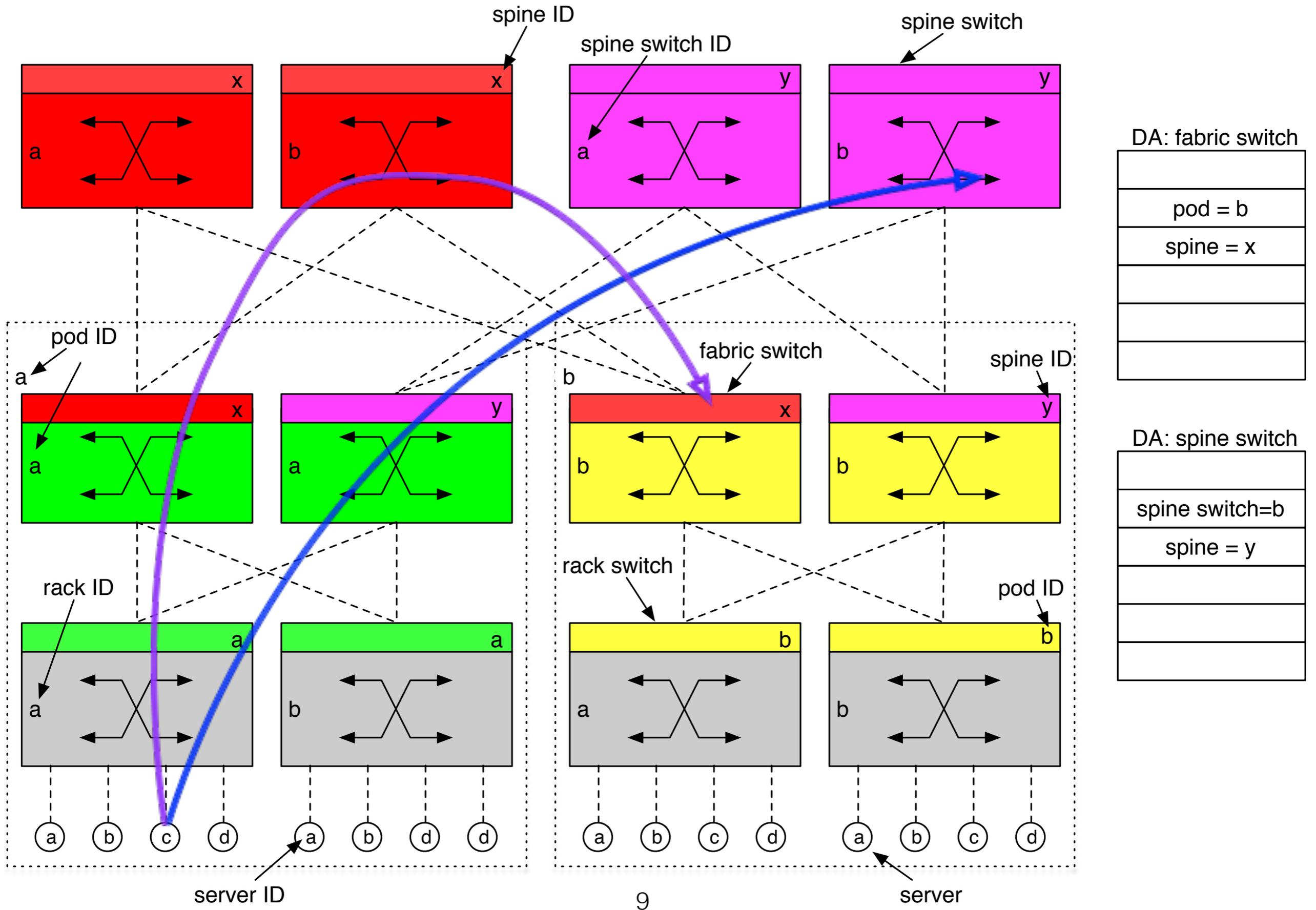
spine switch address



Note 1: server and rack switch format is similar to data-plane address

Note 2: each address format needs to be distinct (in header or otherwise)

DA-embedded Source Routing: Control Frames



Scaling

This partitioning would support, for example:

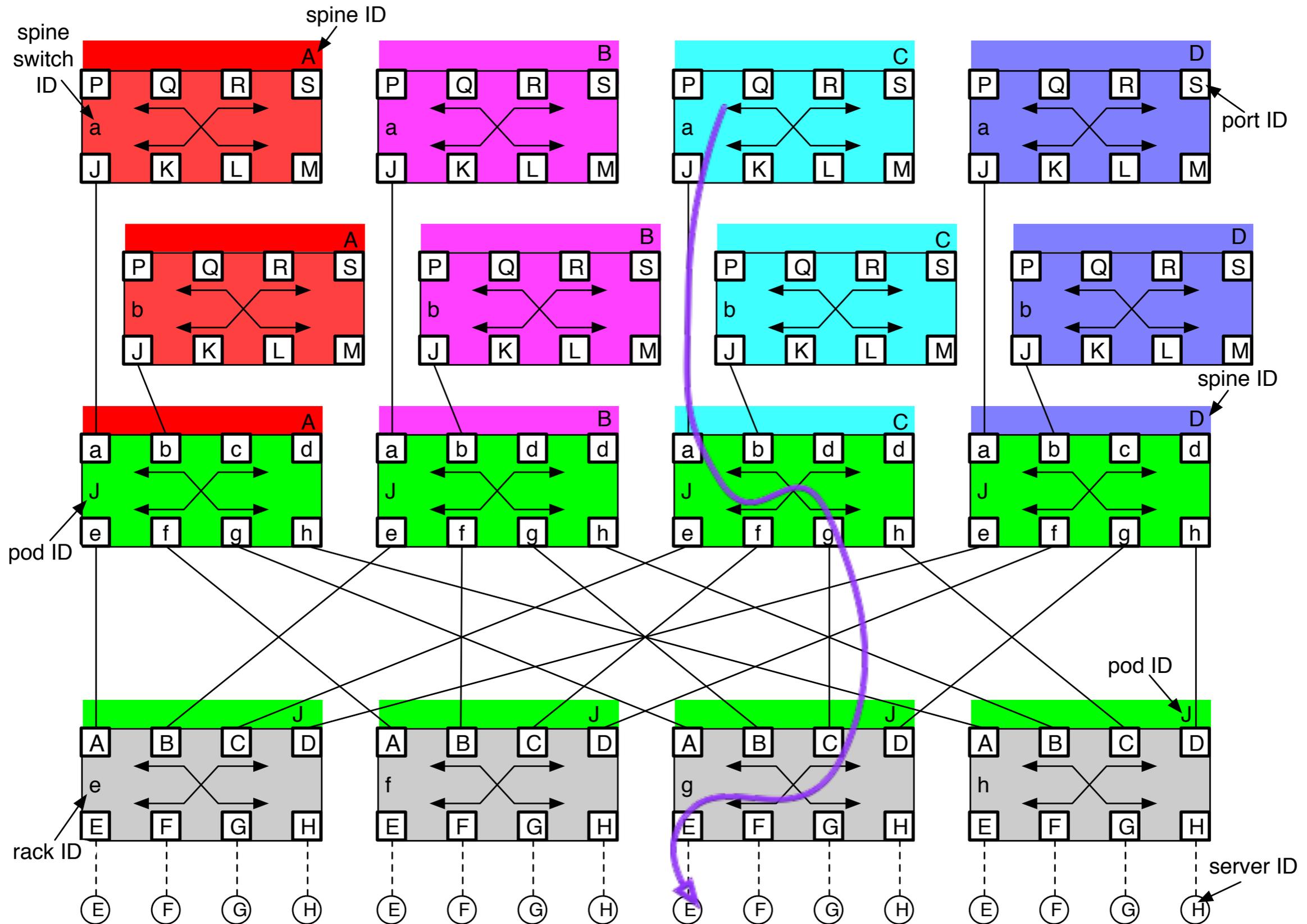
- 256 spines
 - ➔ Facebook uses 4
- 256 spine switches per spine (256 ports per spine switch)
 - ➔ Facebook began with 12, scalable to 48
- 256 pods
 - ➔ Facebook lets this evolve with demand
 - ➔ seems to use 96-port switches; that limits pod count to 96
- 256 racks per pod
 - ➔ Facebook uses 48
- 256 servers per rack
 - ➔ Facebook has been reported to use 48
- 2^{24} (>16M) total servers
 - ➔ Facebook topology “capable of accommodating hundreds of thousands” of servers
- 256 VMs per server
 - ➔ could be more, according to flow ID assignment
- 8 bits per MAC address left over for differentiation of data-plane flows
- Conclusion: this partition allows scaling into the foreseeable future
- Full scale-out uses 256-port spine switches; 512-port fabric and rack switches
 - ➔ This partitioning may be over-dimensioned.

Forwarding Tables

- Flow-zone switching is essentially source-routed, with the route (or equal-cost equivalents) embedded in the DA.
- Switches need not learn or know any MAC addresses.
- Switches need to maintain port mapping tables:
 - spine switch:
 - pod ID => port
 - fabric switch:
 - (local) rack ID => port
 - spine switch ID => port
 - rack switch:
 - spine ID => port
 - (local) server ID => port
- With special connectivity and address assignment, no such mapping tables are required, and the routing is stateless.

Stateless Forwarding

assign zone IDs to match port IDs



Address Assignment

- Presume each spine switch knows that it is a spine switch.
 - Spine switch sends a message to each live port, saying:
 - your pod ID is <outgoing spine switch port ID>
 - Fabric switch receives those messages from spine switches
 - identifies those ports as spine switch ports
 - confirms that all received messages are consistent, with the same pod ID
 - identifies itself as a fabric switch
 - replies to each spine switch, saying:
 - your spine switch ID is <outgoing fabric switch port ID>
 - sends a message to each other live port, saying:
 - your pod ID is <pod ID> and your rack ID is <outgoing fabric switch port ID>
 - Rack switch receives those messages from fabric switches
 - identifies those ports as fabric switch ports
 - confirms that all received messages are consistent, with the same pod ID
 - identifies itself as a rack switch
 - replies to each fabric switch, saying:
 - your spine ID is <outgoing rack switch port ID>
 - sends a message to each other live port:
 - your pod ID is <pod ID>, your rack ID is <rack ID>, and
 - your server ID is <outgoing rack switch port ID>
 - Fabric switch receives messages from rack switches
 - identifies those ports as rack switch ports
 - confirms that all messages from rack switches are consistent
- ➔ Note: Could use, e.g., no-relay DA (01-80-C2-00-00-00 to 01-80-C2-00-00-0F).
- ➔ Note: Above, the identifiers may be just the 7 least significant bits of the port IDs.

Summary

- Flow-zone switching provides efficient Layer 2 routing in the folded Clos or fat-tree architecture of the modern data center.
 - ▶ can be adapted to similar structures
- The Flow-Zone Data-Frame Address Format and Flow-Zone Switch-Frame Address Format embed source routing into addresses.
- Separate data plane and control plane address formats and routing rules.
- Method is loop-free, each hop advancing the frame toward the destination.
- Frames are in standard format and can be bridged normally, with full VLAN transparency.
- Flow identification fields in the addresses can steer behavior in the network and at endpoints.
- The described partitioning scales beyond foreseeable data center network dimensions, using the 48-bit local address format.
- The method can be implemented so that it is essentially stateless, without forwarding tables in the switches.
 - ◉ Switch needs to remember only its own identity.
 - ◉ Stored flow state may be used to improve performance.
- An algorithm to assign addresses for stateless operation has been described.