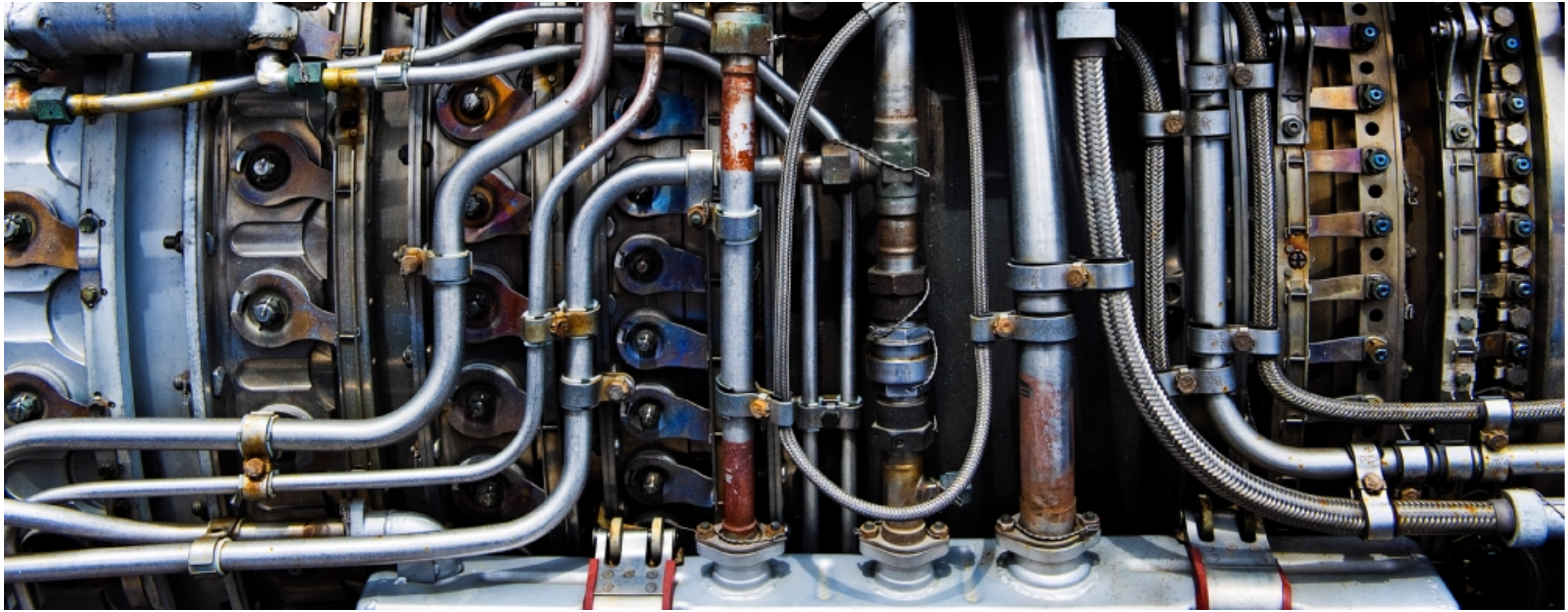
WELCOME



A LOOK AT THE NEW DE-CIX PLATFORM DESIGN

Oliver Knapp, Alcatel-Lucent

DENOG5 - November 2013

Alcatel·Lucent

AGENDA

1. **Working Model and Limitations of Standard IXP Designs**
2. MPLS as a Solution to Scale
3. DE-CIX Setup

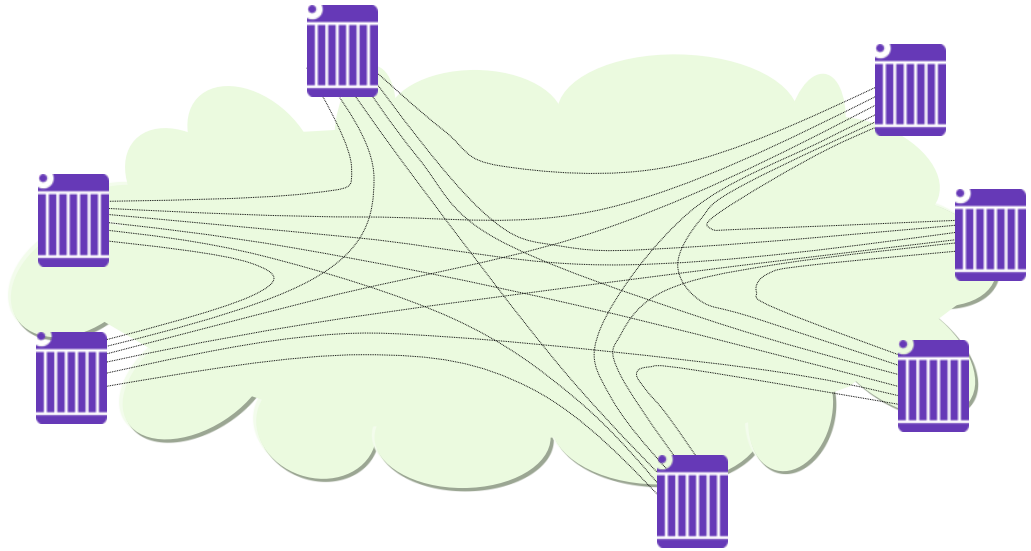
CHALLENGES FOR AN INTERNET EXCHANGE

- Lots of L3 IP Peering customers connected to a big L2 switching layer
- Just “one big switch in the middle” is gone already for some time as a solution
  - Achievable port number and single port bandwidth go reciprocal per chassis
  - At some point, the biggest single switch you can buy is still too small
- Site diversity as a requirement also kills this design model
  - long-lining customer connections wastes bandwidth – some share of traffic could stay local at a sub-site

GOING MULTI-SITE: FLAT MODEL

- How to maintain and scale any-to-any with distributed switches?
- Flat model: build any-to-any links between multiple switches
  - Scaling problem: only fixed flavours of link bandwidth available
  - Increase bandwidth usually by factor ten steps only
  - The more links, the more unused capacity builds up (granularity)
  - Take care of possible loops in that topology

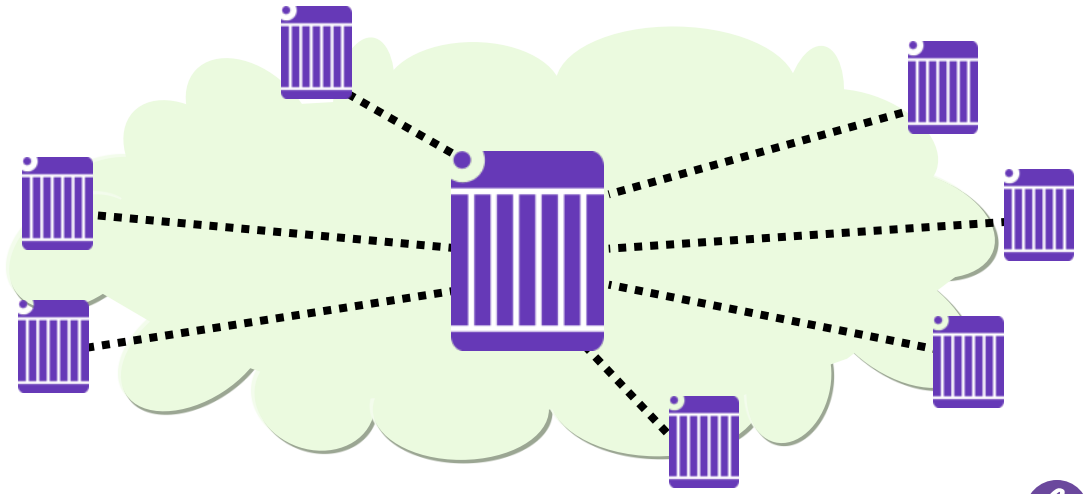
- Solves the multi-site problem
  - But at a high cost for wasted bandwidth (on both switch ports and physical links)
  - Assumes you can buy the required port count per switch at all – including customer ports



GOING MULTI-SITE: HIERARCHICAL MODEL

- How to maintain and scale any-to-any with distributed switches?
- Go hierarchical: split the exchange into an access and a core layer
  - With some luck, some share of traffic stays “local” in the aggregation layer
  - But for the rest, you still need fat interconnects via the core
  - These few fat interconnects in total waste fewer bandwidth than many full mesh links
  - Inherently loop-free

- This also solves the multi-site problem
  - Comparably efficient use of bandwidth
  - But again, how big can you buy that single core switch in the end?
  - Doesn’t fly anymore if the biggest switch on the market is still too small
  - What happens if that single core switch fails ?



WHAT SCALING OPTIONS EXIST FOR THIS HIERARCHICAL MODEL?

1. Increase bandwidth of interconnection links
   - Has a technological end of scaling: 100G is biggest available technology today

2. Bundle more links together
   - LAG link bundles work as single logical link – still no loop problem
   - But: each switch implementation has a maximum number of links that can be bundled into a LAG
   - Number of ports per chassis is finally intrinsically limited by chassis size
   - DE-CIX is already beyond that maximum for a single core chassis since a long time ago

The classic "Fat Tree" approach comes to an end here!

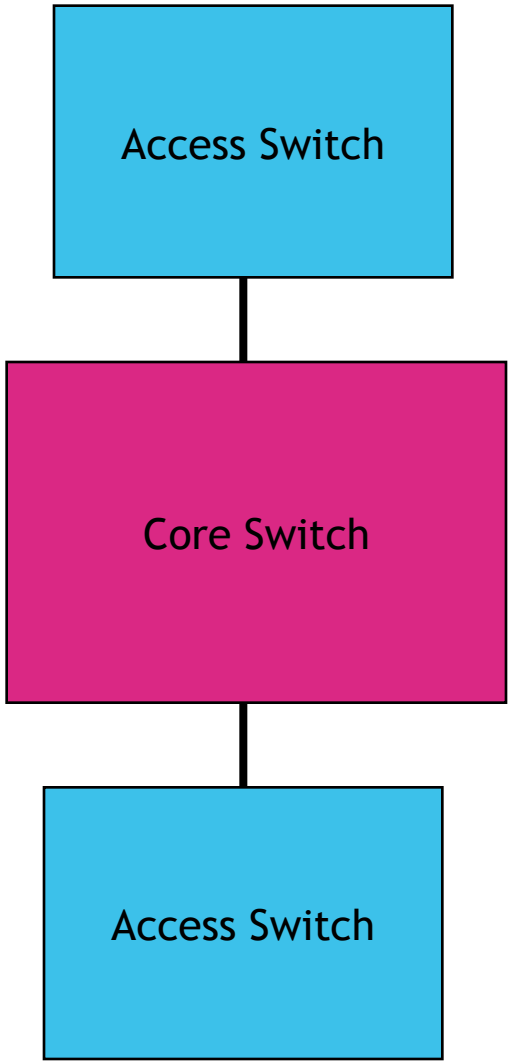
WHAT OTHER APPROACHES EXIST?

- Check the classic telephone switching world: quite similar problems 60 years ago
  - Capacity problems in single crossbar telephone switches
  - “Clos network” as a layered solution (1953)

- In data networking: today commonly referred as “Spine-Leaf architecture”

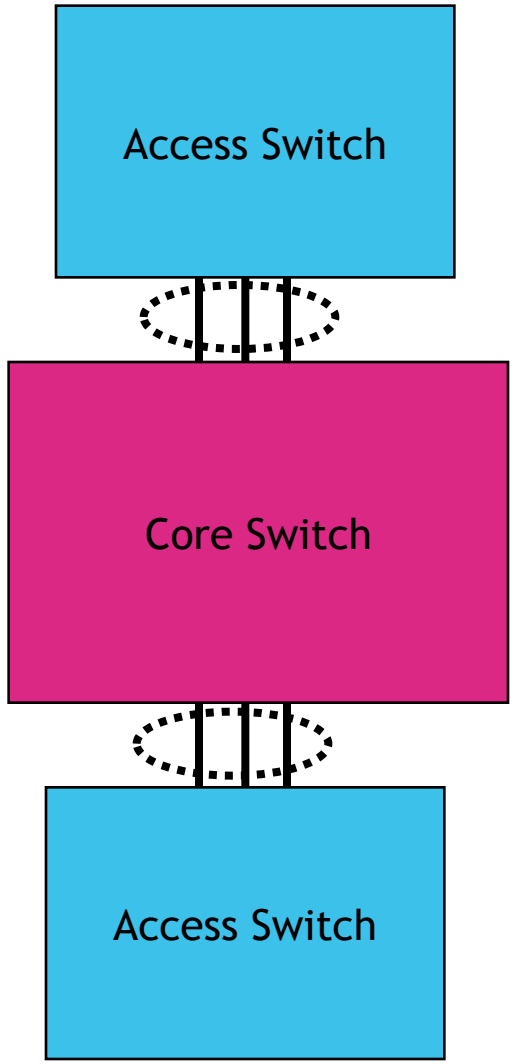
- Principle: go highly parallel, if purely hierarchical approach doesn’t scale anymore
  - Proven design for internal switch fabrics of networking gear
  - Common solution for large data center switched networks
  - Supercomputer clusters leverage this design as well

A SIMPLE CONNECTIVITY MODEL



- For simplicity, we just look at how two access switches communicate to each other via the core
- There is no limit in this model for adding more access switches, other than total core switch port capacity
- Between the two access switches, the core is always just one “hop”
- All plain, switched Ethernet
- Nothing new - you know it already

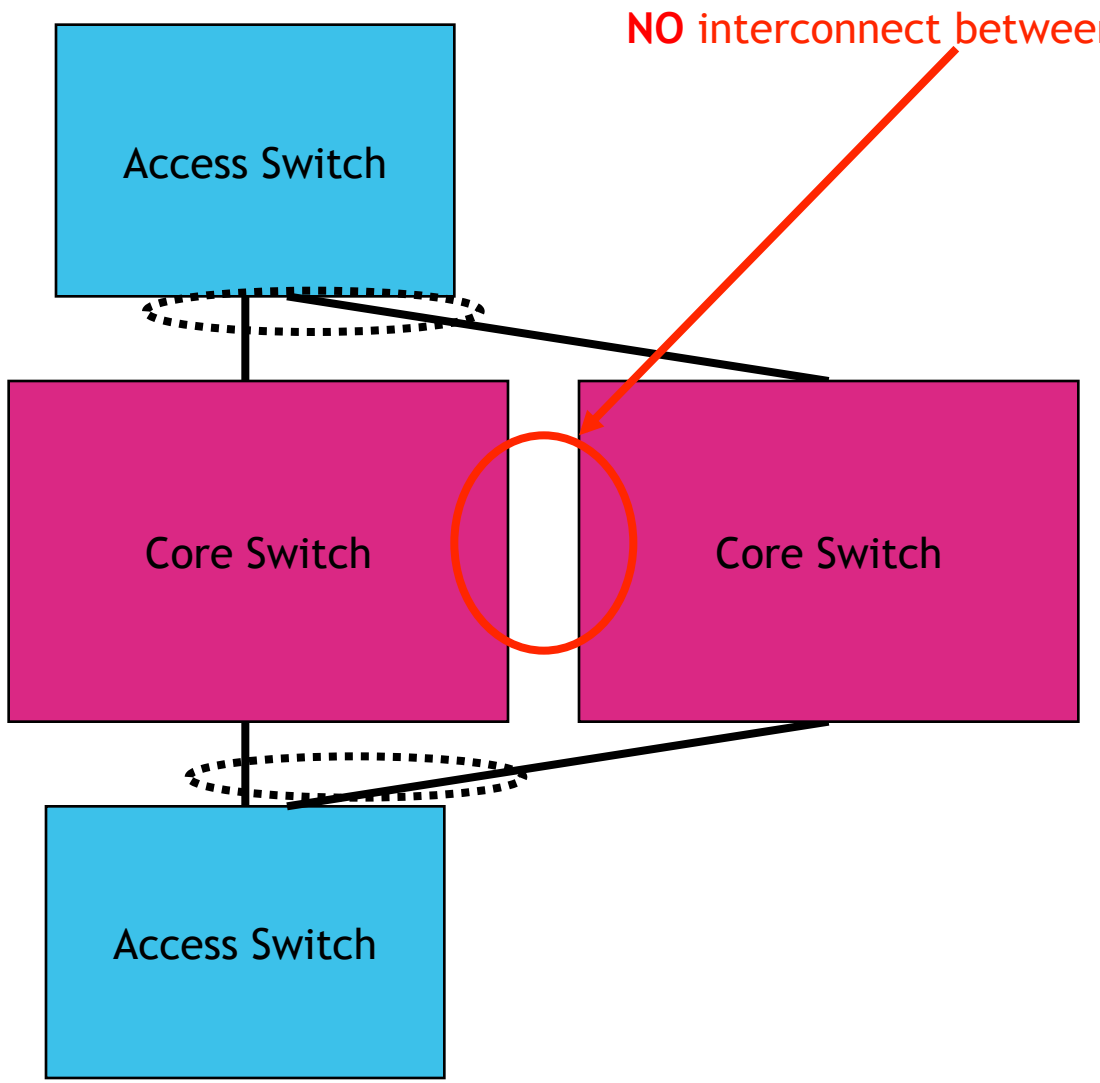
SCALING: OPTION 1 - ADD BANDWIDTH



As already discussed:

- As long as higher bandwidth physical links are available, just use those instead
- If you already use the biggest links you can buy, bundle more of them together into one logical link
- “Link aggregation”
- Scaling limit is port count per core chassis
- Also a well-known approach

SCALING: OPTION 2 – GO PARALLEL, ADD ANOTHER CORE



- Introduce a second core chassis
- Split the uplinks from the access switches over both cores
- Ensure you balance traffic load reasonably
- This was pretty much DE-CIX's LAG workaround with the old core switches
  - Not really covered by LAG standard
  - But works in practice by hashing in the access, and MAC learning in the core switches
- Final limit now: LAG scaling in the access switches

CHALLENGES WHEN GOING MULTI-CORE

- Using logically separate links does not work: loops would be created
- Use single logical links through different cores?
  - As mentioned, works in practice with LAG
  - But not a clean solution – slightly outside of standards
  - Limited by access switch LAG scaling

- Alternative standardized solution: Equal Cost Multipath Routing

...Routing ???

...In plain L2 Ethernet ???

→ That might sound a bit weird at a first glance...

AGENDA

1. Working Model and Limitations of Standard IXP Designs
2. **MPLS as a Solution to Scale**
3. DE-CIX Setup

WAYS TOWARDS ECMP FOR LAYER 2 ETHERNET

- In the Layer 3 routing world, some problems of the Layer 2 world are already nicely solved:
  - Loop prevention
    - No flooding
    - Only best path forwarding
    - Hop count limit
  - Best path selection
    - Routing protocols
  - Load sharing
    - Multiple, equal cost paths (ECMP)

- We can inherit these features by encapsulating Ethernet traffic into “something based on L3 transport capabilities”

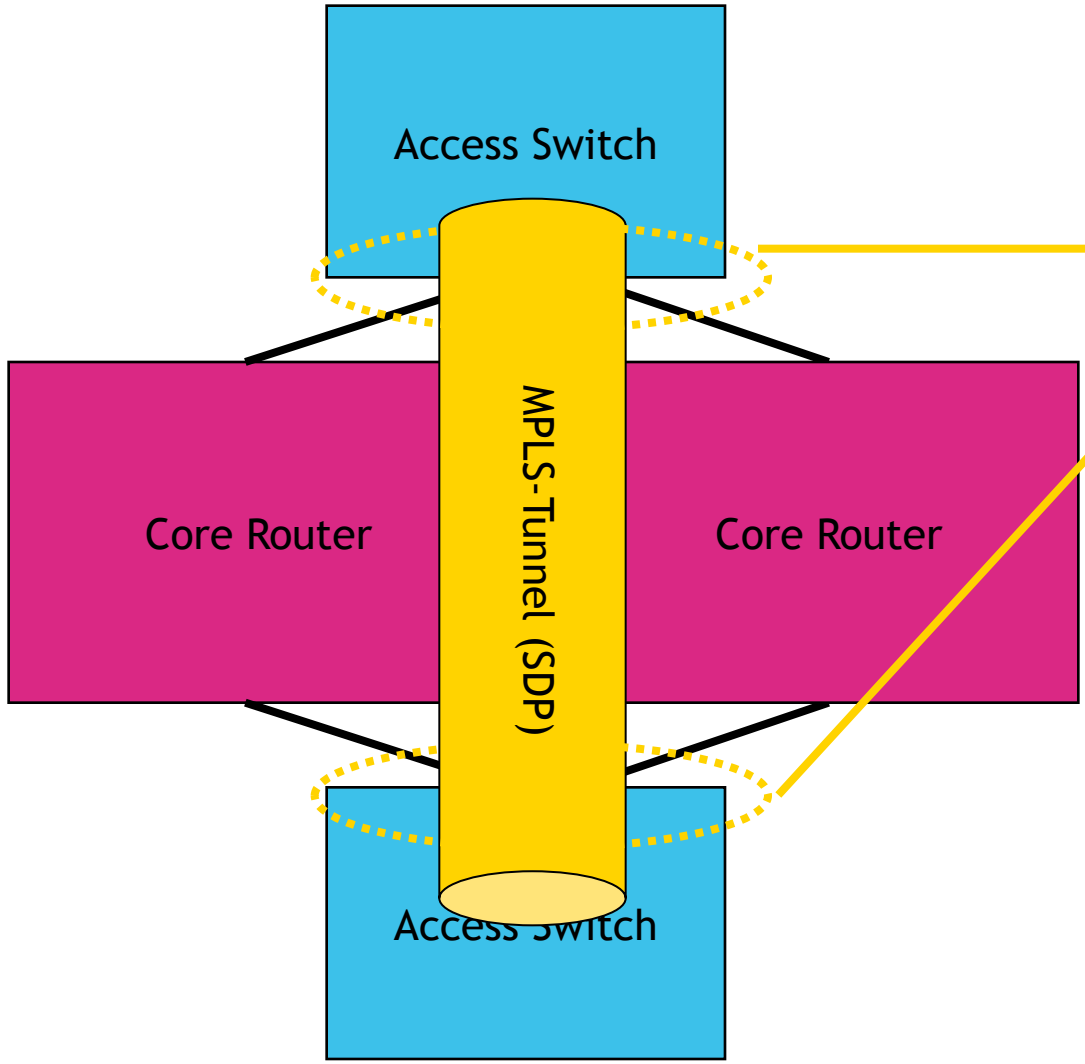
Proposed solution approach: MPLS

MPLS FOR LAYER 2 ETHERNET

- MPLS is a widely known and field proven way to transport Ethernet
- MPLS very nicely abstracts logical paths (tunnels) from physical links
- For what we need here, the most simple MPLS flavour is sufficient: LDP
  - LDP follows the IGP by nature: ECMP in IGP means ECMP also for LDP
  - No TE capabilities required nor available in our case (resilient links = wasted bandwidth)
  - Very simple configuration and operations

- Both point-to-point and multipoint-to-multipoint L2 services are well defined
- Obviously, the MPLS equivalent of an Ethernet switch is a VPLS

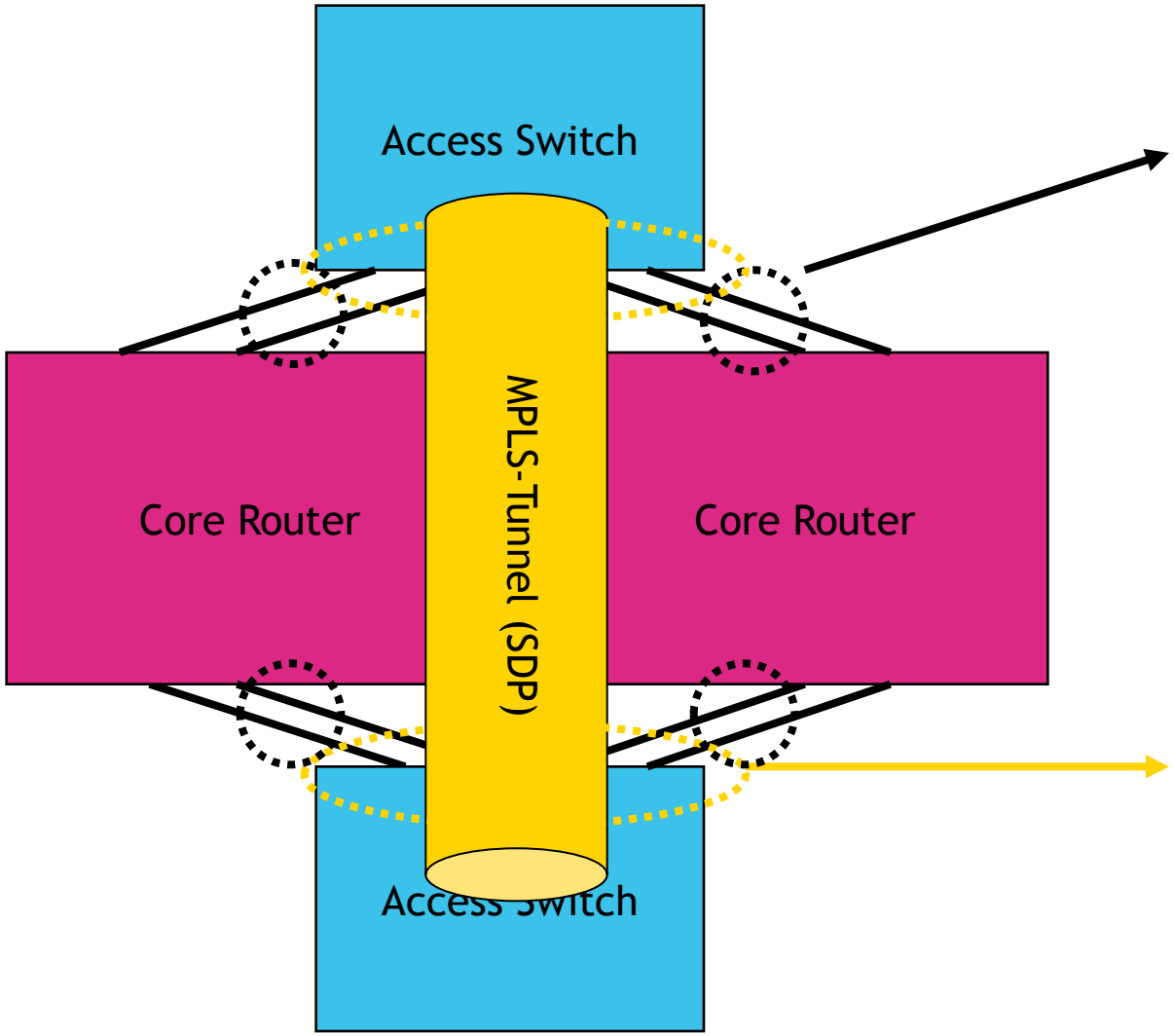
ECMP FOR MPLS



When two diverse, equal cost IGP routes exist between two access switches, traffic load on the resulting SDP* will be balanced equally over these two paths as well when doing MPLS/LDP

*= a Service Delivery Path is the Alcatel-Lucent term for “any flavour of an existing, usable transport tunnel” within our overall data model

PUTTING IT ALL TOGETHER



1) Use LAG bundles to increase capacity on the underlying network links between access and core. This creates single logical network IP/MPLS interfaces and links with aggregated bandwidth.

2) Use MPLS over ECMP to create parallel paths between the access switches over both core switches, increasing overall available bandwidth.

This is still one logical tunnel only, having two alternative logical, and four separate physical paths.

WHAT IF THAT'S STILL NOT ENOUGH ?

- Add another parallel core router – design is not limited to two cores only
- Final limitation then: number of uplinks out of an access switch
- Solution: add a second (third,...) non-interconnected access switch per site
  - traffic between customers on different switches then trombones through the core and back
  - not a real latency problem in a metro area
- This approach scales VERY far...
  - in theory, you could use up to half of the available ports of an access switch as uplinks going to one separate core router each
  - maybe even a bit less uplink capacity is required, as some whare of traffic would stay local within the access switch
  - you can easily imagine how far that would go using big chassis and 100G links

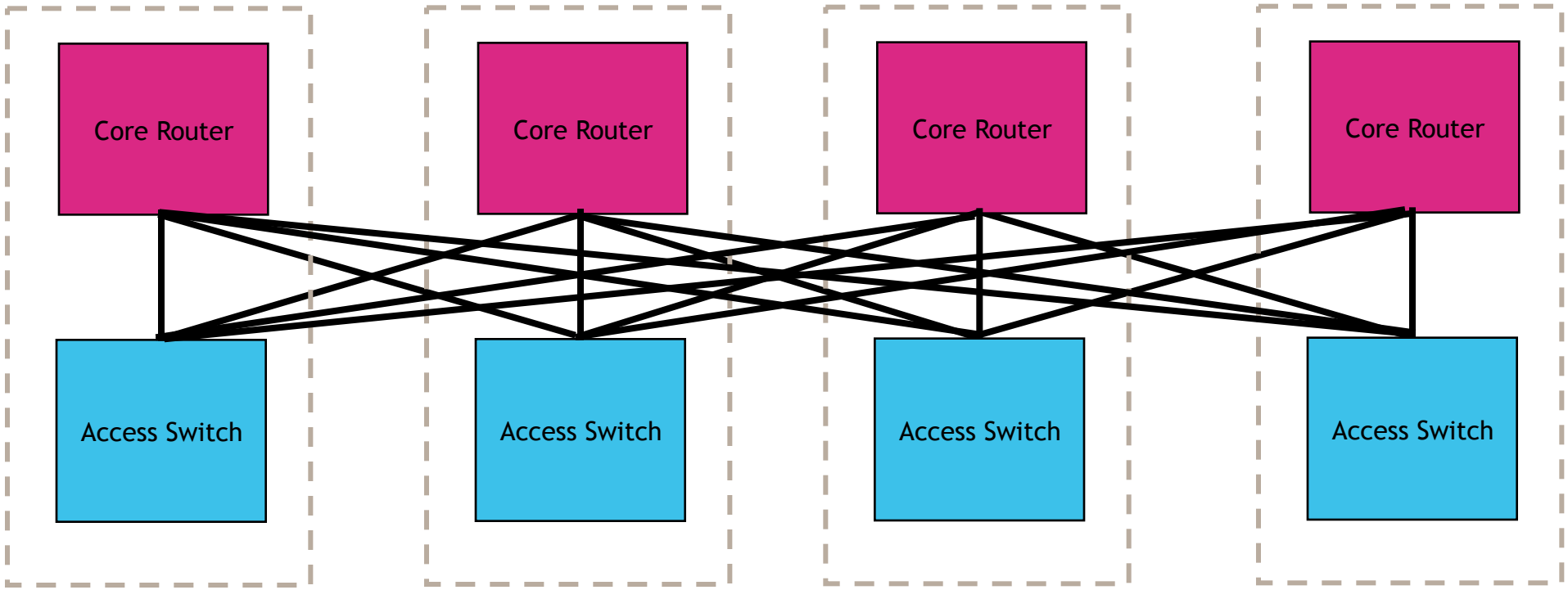
SUMMARY: MPLS-BASED SPINE-LEAF DESIGN

- LAG builds “fat pipes” to run MPLS between access and each core switch
- MPLS then together with ECMP allows building parallel paths for each access switch pair through all the core switches, and distributes load over all these paths
- These two levels of bandwidth bundling overall result in a multiplication of the respective bundle scaling that can be achieved by each LAG and ECMP alone
- The design theory itself is neither limited in the number of access switches nor core switches
- In practice, there is limits for both links within a LAG and also ECMP paths
- Single port bandwidth and port count per chassis are a technical limit

AGENDA

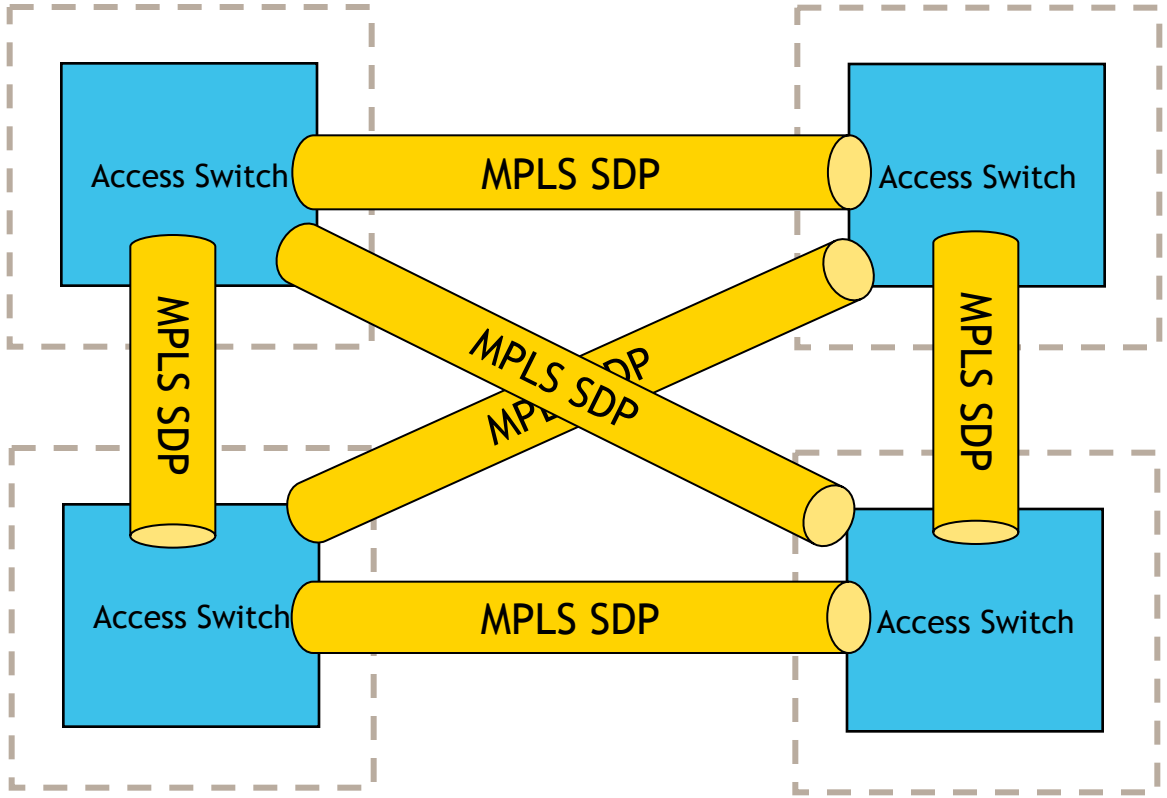
1. Working Model and Limitations of Standard IXP Designs
2. MPLS as a Solution to Scale
3. **DE-CIX Setup**

DE-CIX PHYSICAL DESIGN



- DE-CIX uses four parallel core routers today
- Each core switch has a colocated edge switch (direct back-to-back link, all others via WDM)
- Full mesh: all edges connect into all cores
- For smaller locations, backhaul switches are used hanging off the edge switches (not shown)

DE-CIX LOGICAL DESIGN



- Differs greatly from the physical topology: Logical full mesh VPLS design, inherently loop-free
- Core doesn’t appear, as it is part of the MPLS transport (P-router only, service unaware)
- Still, bandwidth of this full mesh is shared over all the “fat pipe” physical links in parallel

QUESTIONS ?

www.alcatel-lucent.com