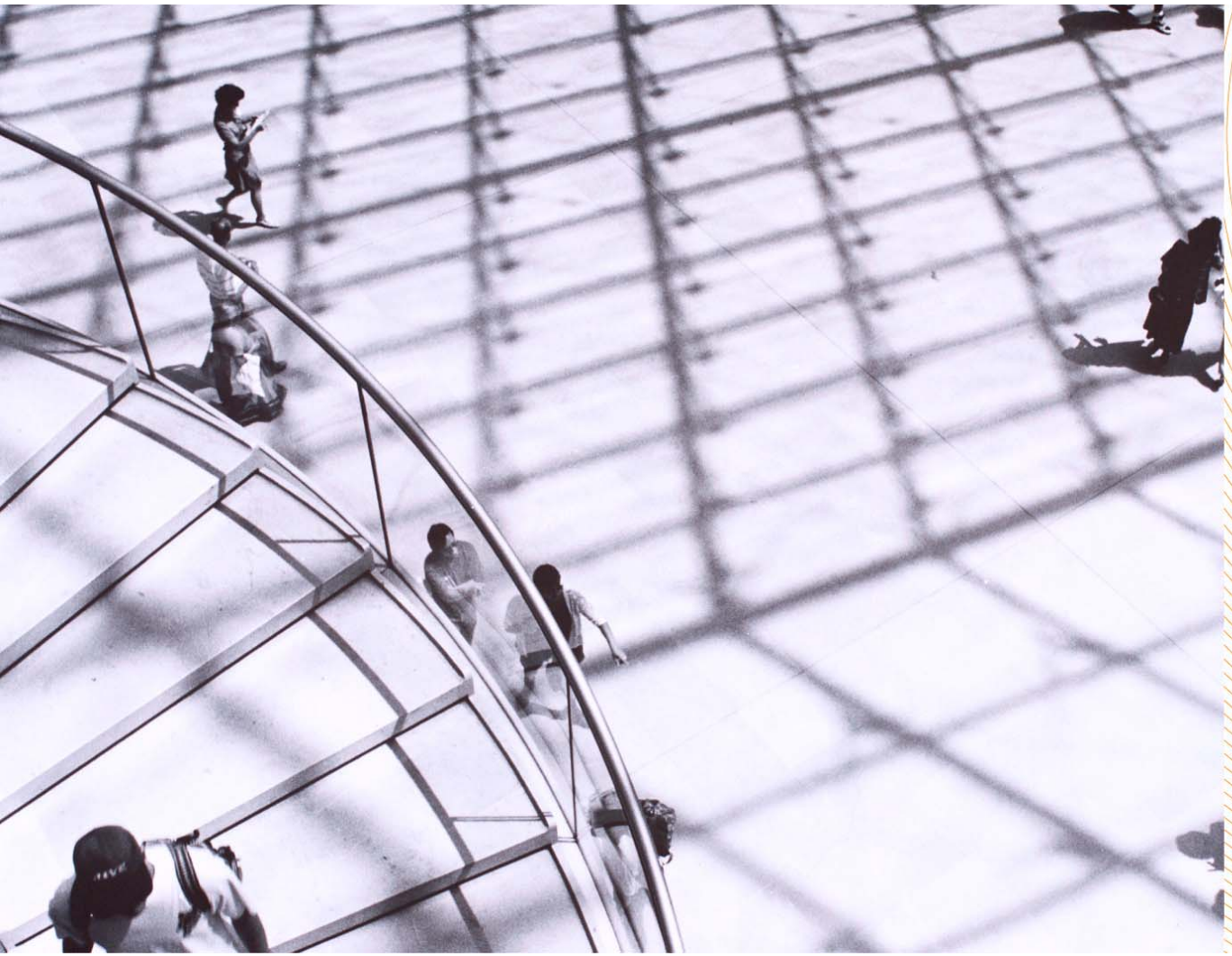
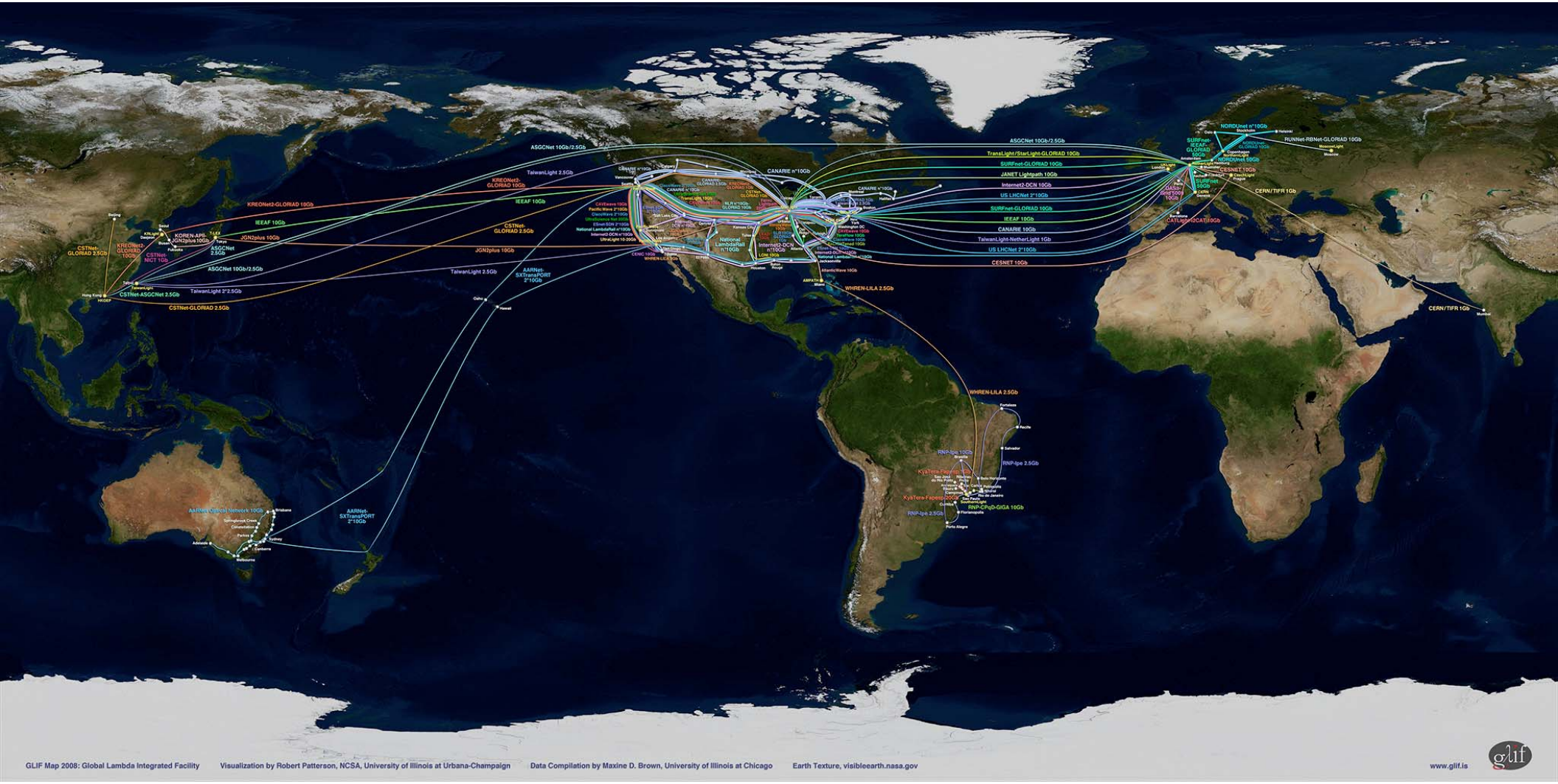


Black box approach in NetherLight

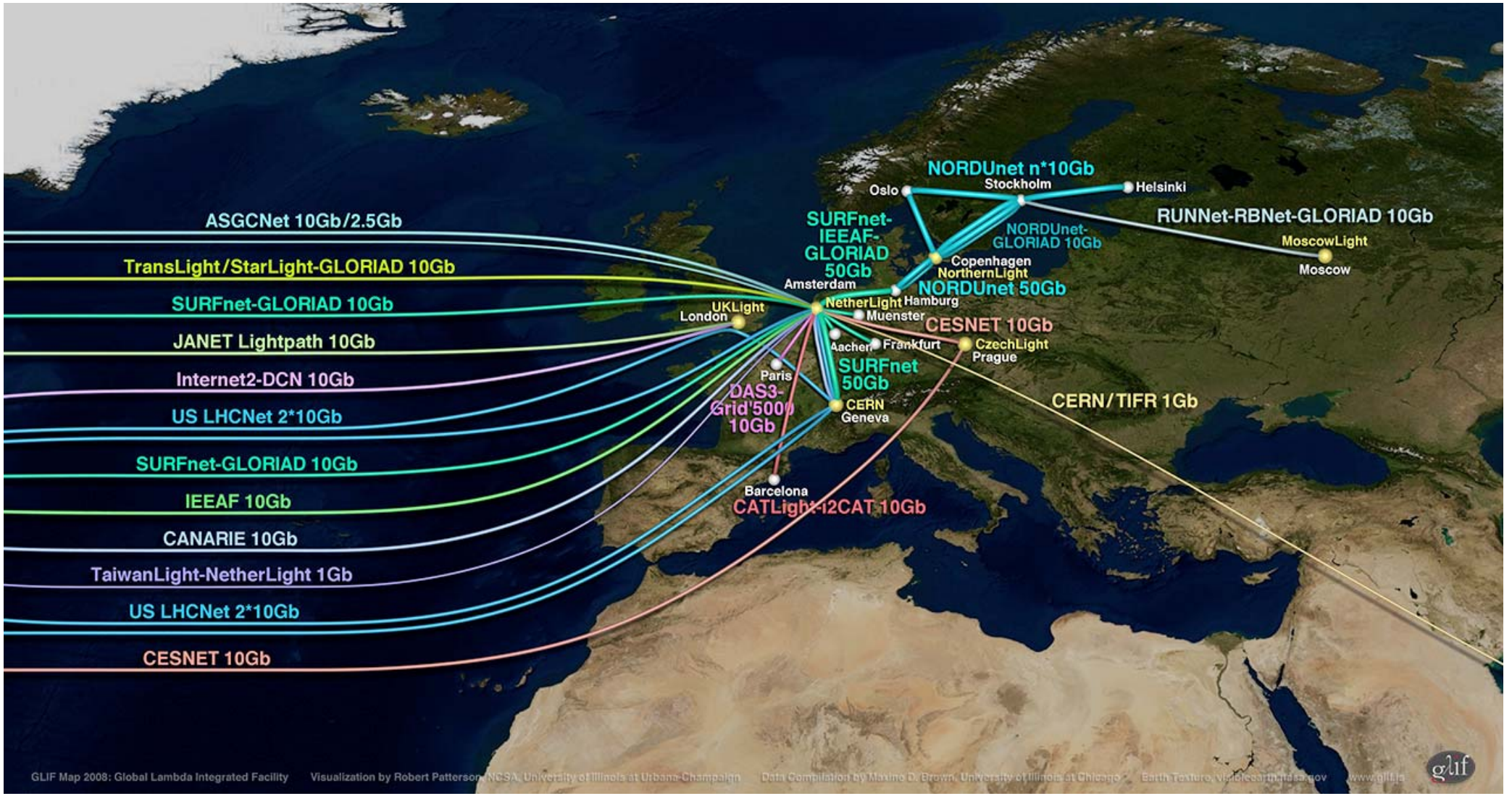
Wouter Huisman

December 2009

GLIF community
Global resources

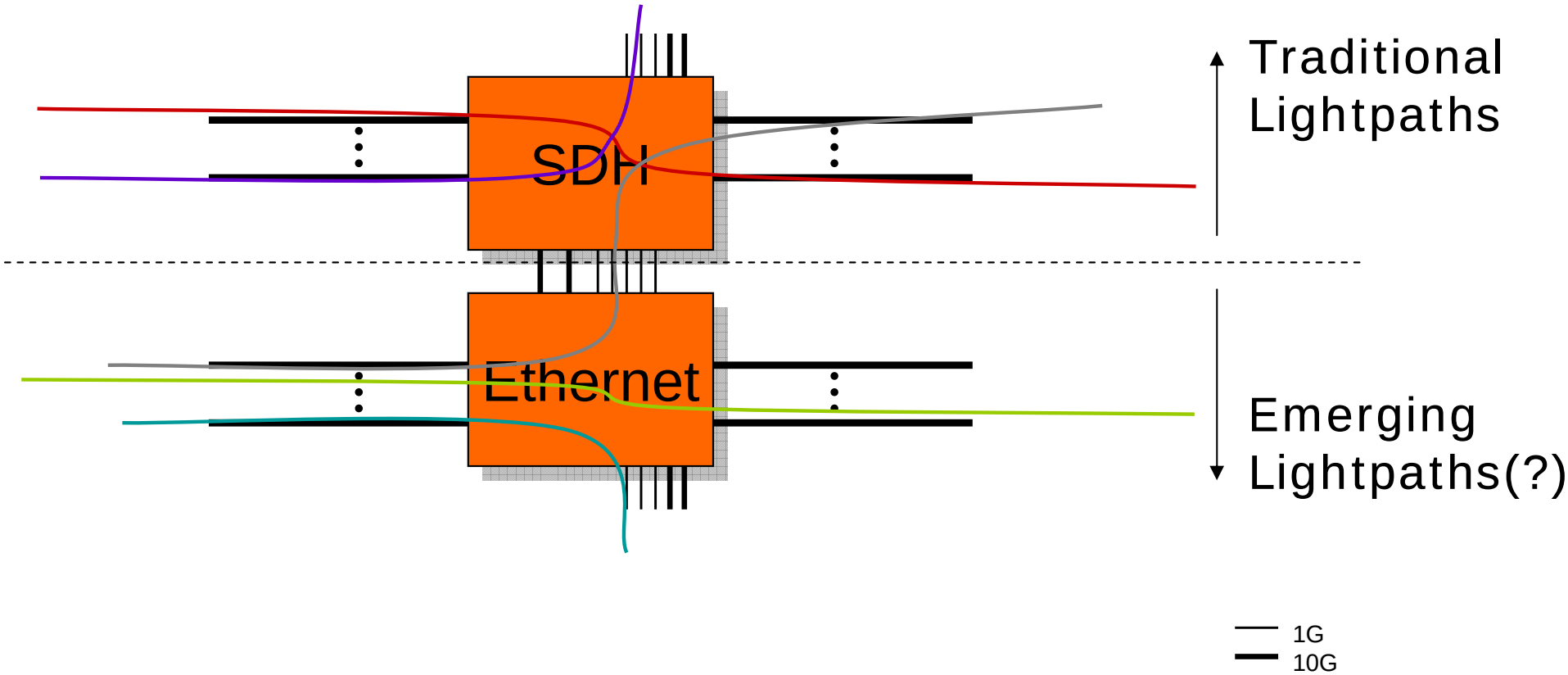


GLIF community
Global resources



NetherLight Services

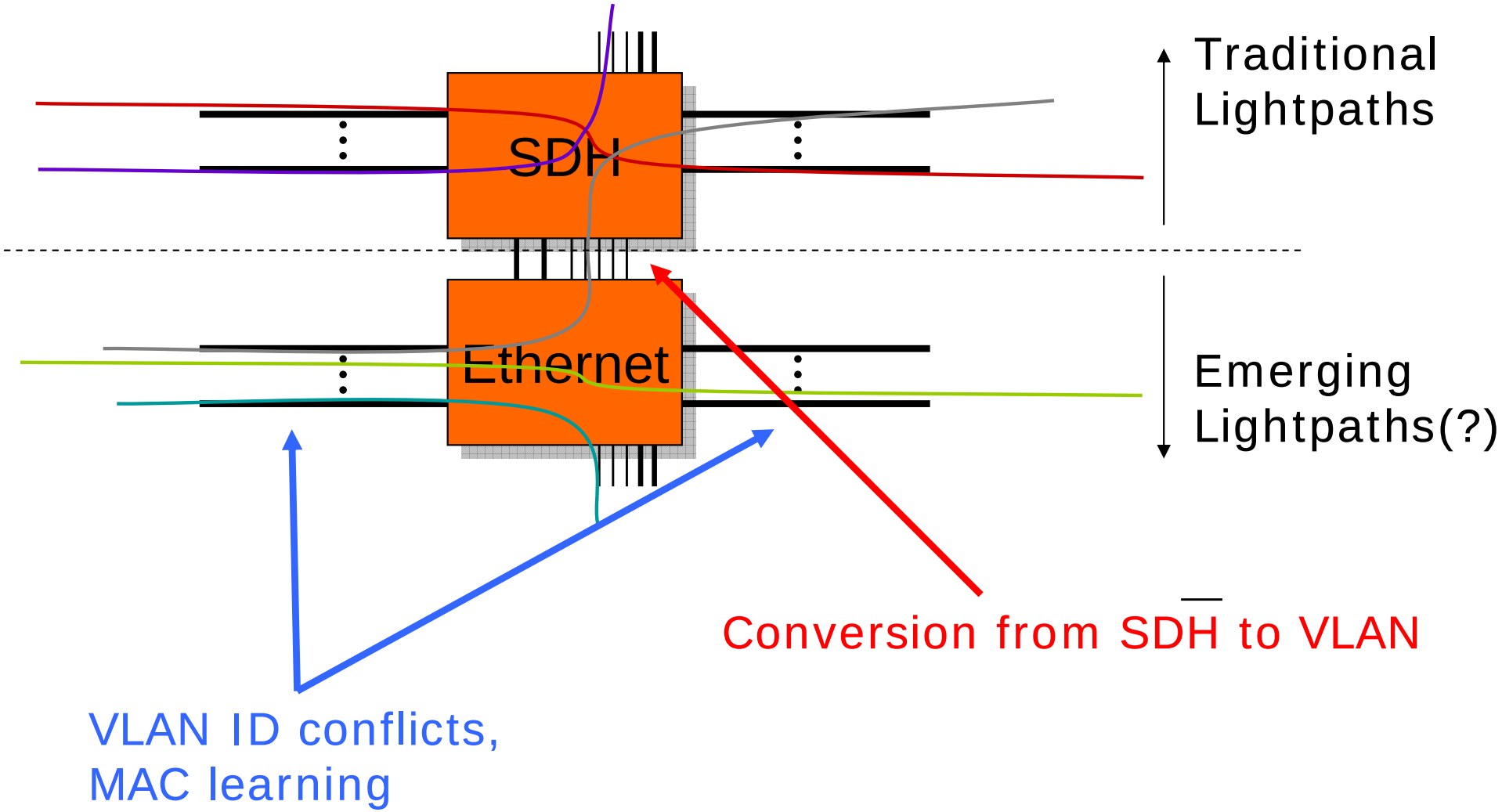
- Lightpath using SONET/SDH transport layer
- VLAN based connections using Ethernet layer 2
- Or combined service



Why Ethernet as Transport layer

- All services on NetherLight are Ethernet, no GFP/LEX mapping
- Projects in temporary hibernation mode don't consume bandwidth, minimal effort for GOLE operations
- Statistical multiplexing: better utilisation on transatlantic links (scarce bandwidth)
- Multiple services per client interface (eg. GE port)
- Mesh is allowed, not limited to point-to-point

Focus Areas

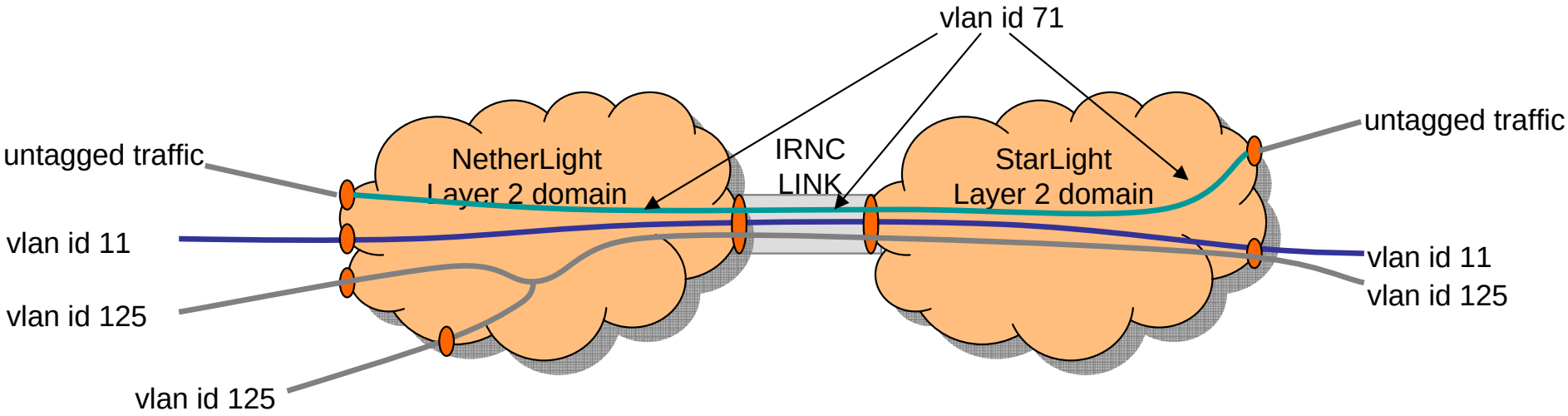

SDH
Ethernet
Traditional Lightpaths
Emerging Lightpaths(?)
Conversion from SDH to VLAN
VLAN ID conflicts,
MAC learning

Current Ethernet services

- Two connection models apply based on VLAN tagging (802.1Q):
  - Tagged traffic originating from an Ethernet network is passed on using the same VLAN tag
  - Untagged traffic originating from a directly connected host, data is mapped to a new VLAN ID.

Best effort services, ideally with no overprovisioning to guarantee performance

Ethernet services current practise



Physical port

Limitations 802.1Q

- Max number of 4095 VLAN Ids may result in stranded VLAN Ids for connected GOLEs
- Finding an available VLAN ID is time consuming since many parties are involved (eg Arecibo – JIVE)
- No clear separation between customer network and provider network
- Backbone MAC table contains all customer MAC addresses
  - Flooding of MAC table

Alternatives for 802.1Q

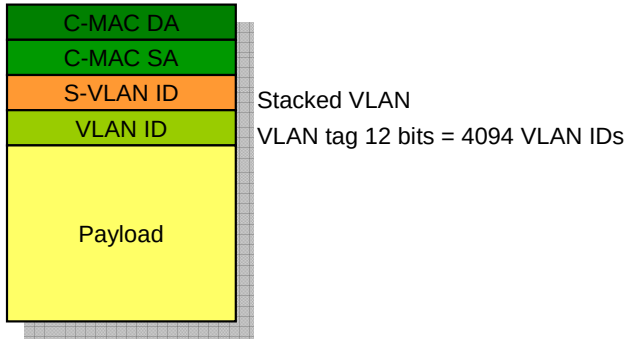
Instead of VLAN tagging, two encapsulation methods could be used:

- QinQ, 802.1ad
- PBB / Mac-in-Mac, 802.1ah

802.1ad Q-in-Q

Q-in-Q is an Ethernet standard defining an aggregating function for multiple VLANs into 1 provider VLAN

- transparently maps any VLAN to a provider VLAN ID
- Customer VLANs are separated from Provider VLANs



Limitations 802.1ad

- Separation between customer network and provider network only on VLAN ID
- Backbone MAC table contains all customer MAC addresses
  - Flooding of MAC table

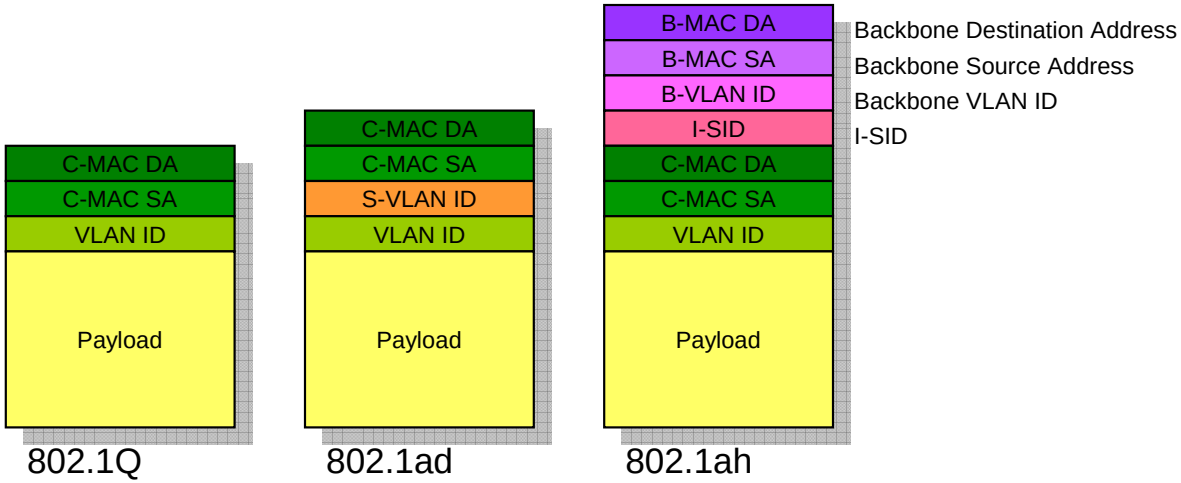
- Aggregation only:

  No flexible switching, since all incoming traffic is mapped to 1 provider vlan → not suitable for NetherLight.

802.1ah / PBB / Mac-in-Mac

Ethernet standard allowing mapping of VLAN IDs to I-SID

- Scalable to 16M services
- Clear separation between customer and provider network
- MAC addresses separation
- Similar feature set as VLAN tagging + more



Limitations 802.1ah

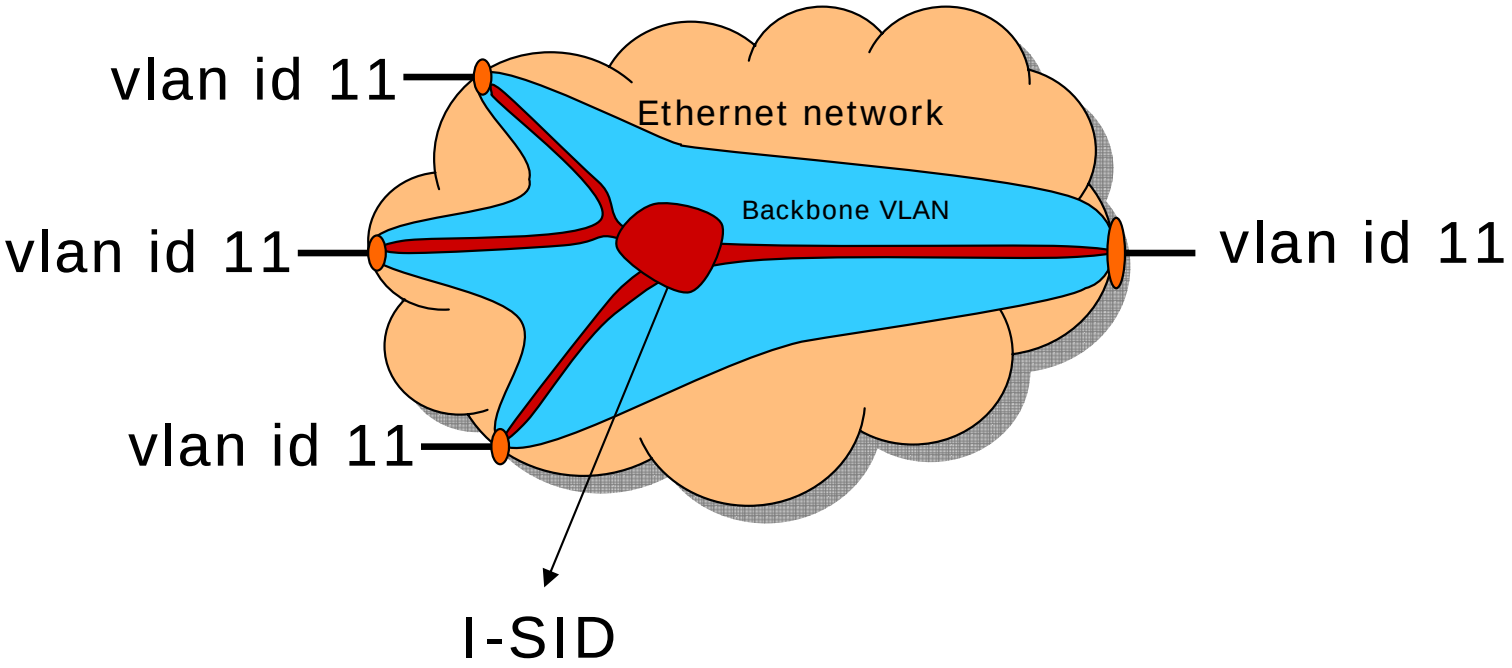
- Slightly larger overhead (22 bytes)
- Bit more configuration work to set up a service
- Not too many vendors support 802.1ah

Setting up Mac-in-Mac

1 - Define Backbone VLAN (one per network)

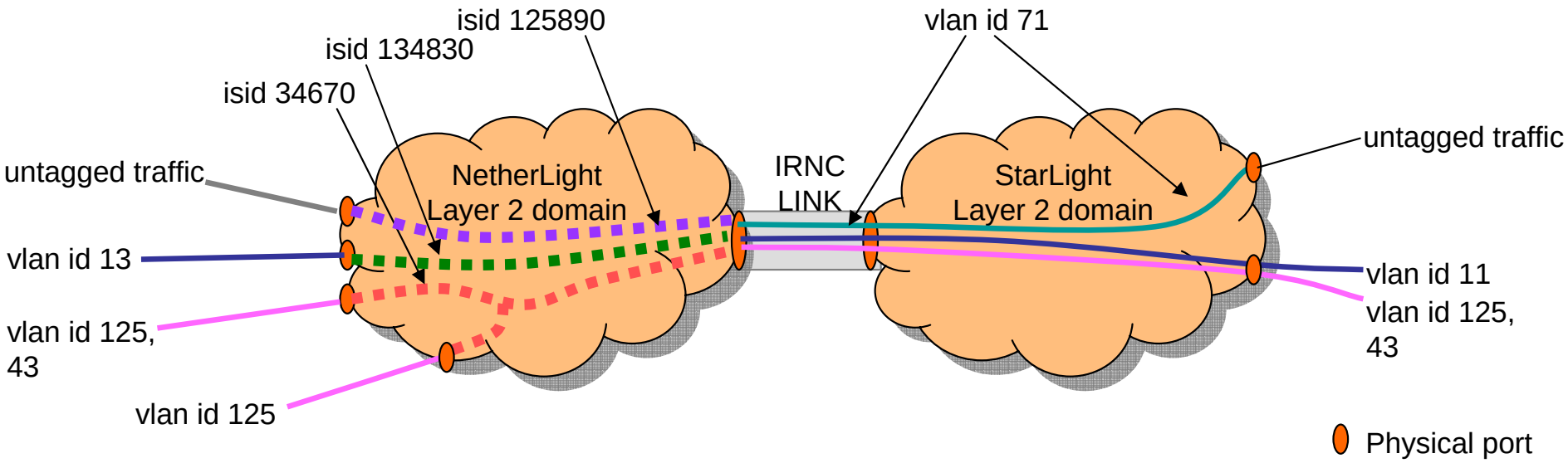
2 - Add I-SID per service instance

3 - Create endpoints for I-SIDs (mapping VLANs) to I-SID



Tested solutions

- Untagged traffic → VLAN
- Transparent VLAN
- VLAN retagging
- Trunk VLANs



Conclusions

802.1ah / PBB brings value to NetherLight

- Flexibility
  - VLAN retagging
  - Customer VLAN agnostic
- Secure
  - MAC address separation
- Scalable
  - 16M services
  - No VLANID stranding

But:

- No replacement for SDH or SONET services

Black Box

- NetherLight supports two data planes:
  - SDH Xconnects and Ethernet VLANs, which only interwork via patch cords.

- Any GE / 10G flow changing between dataplane needs a 1G / 10G interface back-to-back to connect.

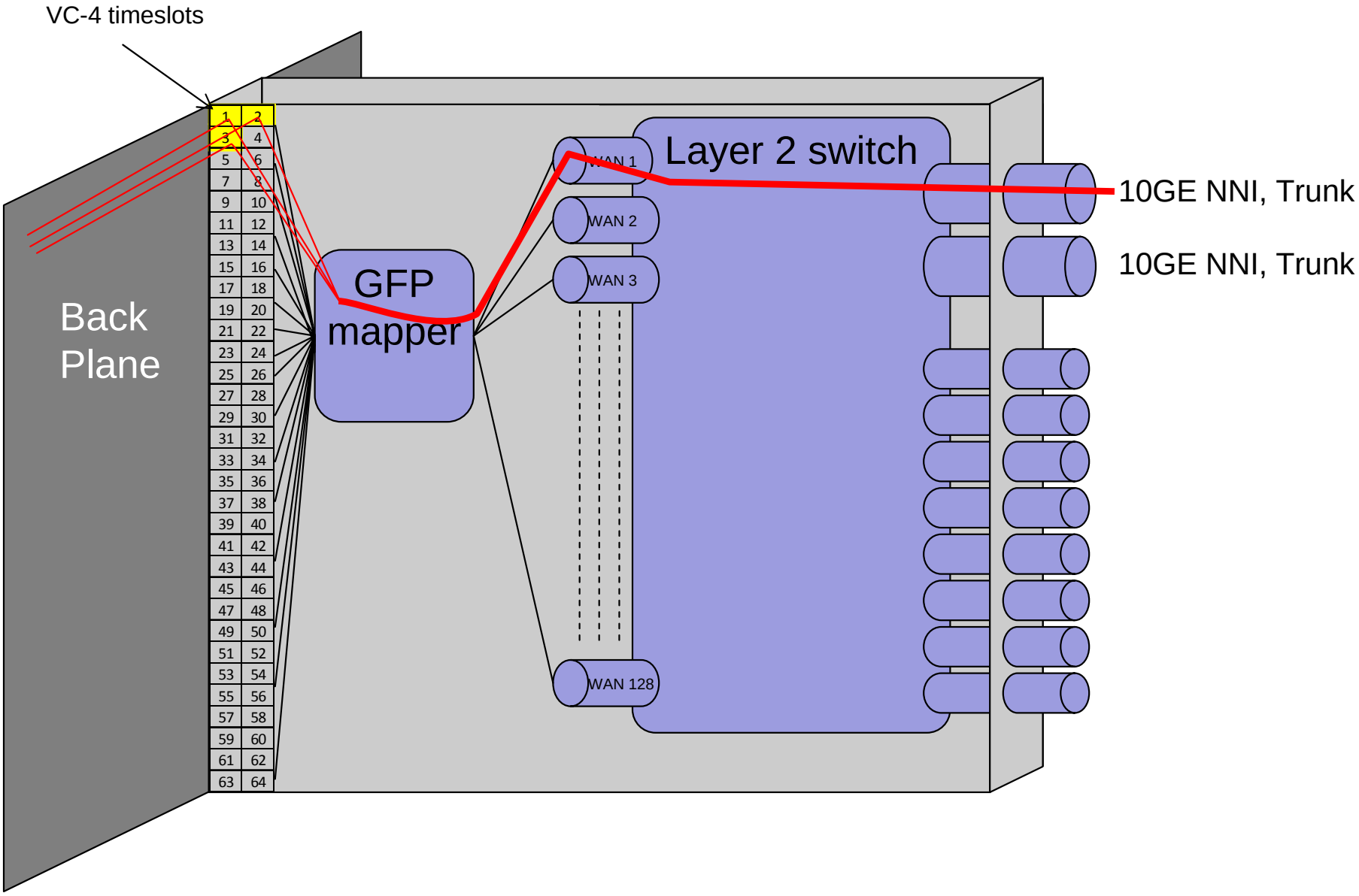
- Internal port cost is high

20G L2SS card

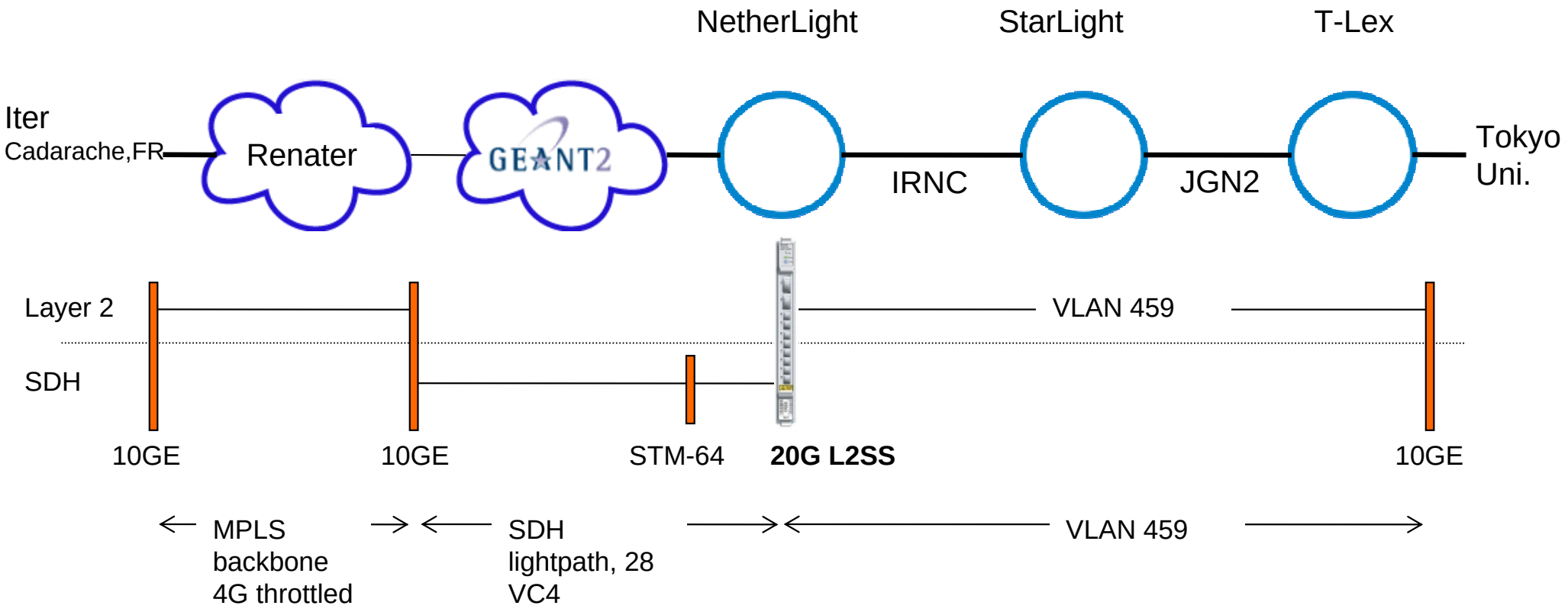
- Supported in Nortel OME6500 from R5.2.1
- Faceplate interfaces supports of 2x 10GE LAN Phy and 8x 1GE ports
- Backplane supports max128 WAN ports using GFP mapping and VCAT
- Backplane access max 10G

- Possibility for circuits >1G

Logical Overview



Tokyo- Cadarache 4Gbs lightpath



Thank you

Wouter.huisman@surfnet.nl

http://noc.netherlight.net

www.surfnet.nl/nl/Thema/netherlight