

# Software Defined Networking for HPC Interconnect and its Extension across Domains

Michael Lang  
Los Alamos National  
Laboratory

Xin Yuan  
Florida State  
University



LA-UR 16-21010

# Focus of our project:

## OpenFlow-style SDN for HPC Interconnects

- Internally: within the interconnect
  - Optimize routing for individual applications and system
- Externally: match external network resources to HPC cluster
  - Remote users, remote resources
  - Storage systems
  - Data Analytics and Visualization clusters
  - Experimental Facilities (MaRie LANL)

# HPC Interconnects: State of the Art

- Existing interconnection networking technology for HPC systems
  - InfiniBand:
    - high bandwidth and low latency
    - Flexible topologies: Fat-tree topology for most operational systems, also support other topologies
    - Simple control: destination based routing - inflexible and inefficient
    - Performance issues when scale up
  - Proprietary technologies:
    - Less flexible topologies
    - Complex control
      - Cray Cascade (Edison):
        - » Dragonfly topology
        - » Global adaptive routing
      - IBM Bluegene:
        - » 3D and 5D torus topologies
        - » Adaptive routing

# HPC Interconnects: State of the Art

- Existing interconnection networking technology for HPC systems
  - InfiniBand:
    - high bandwidth and low latency
    - Flexible topologies: Fat-tree topology for most operational systems, also support other topologies
    - Simple control: destination based routing - inflexible and inefficient
    - Performance issues when scale up
  - Proprietary technologies:
    - Less flexible topologies
    - Complex control
      - Cray Cascade (Edison)
        - » Dragonfly topology
        - » Global adaptive routing
      - IBM Bluegene:
        - » 3D and 5D topologies
        - » Adaptive routing



\*\*Ethernet and SDN/Openflow will be persistent!

# HPC Interconnects: State of the Art

- Existing interconnection networking technology for HPC systems

- InfiniBand:

- high bandwidth and low latency
    - Flexible topologies: Fat-tree support other topologies
    - Simple control: destination-based
    - Performance issues when scaling

- Proprietary technologies:

- Less flexible topologies
    - Complex control
      - Cray Cascade (Edison):
        - » Dragonfly topology
        - » Global adaptive routing
      - IBM Bluegene:
        - » 3D and 5D torus topologies

## KNL with Omni-Path™

Omni-Path™ Fabric integrated on package

First product with integrated fabric

Connected to KNL die via 2 x16 PCIe\* ports

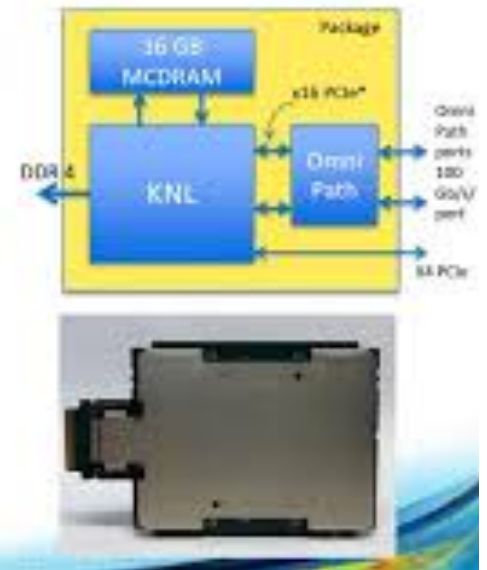
Output: 2 Omni-Path ports

- 25 GB/s/port (bi-dir)

### Benefits

- Lower cost, latency and power
- Higher density and bandwidth
- Higher scalability

\*On package connect with PCIe semantics, with HPC optimizations for physical layer

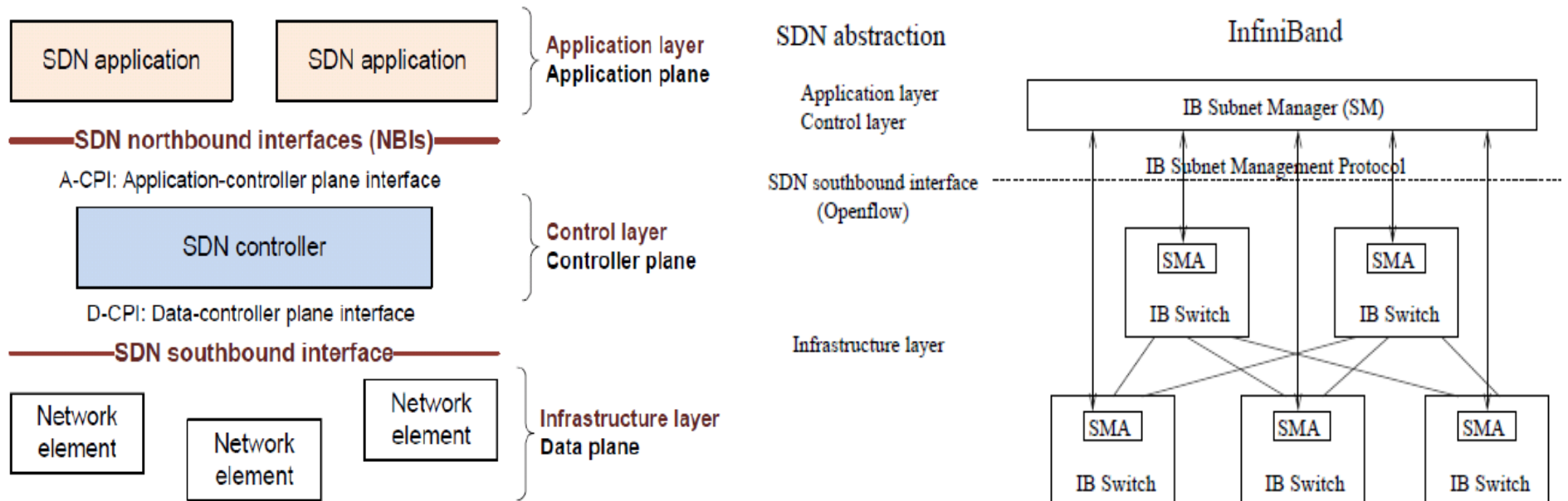


# HPC Interconnects: State of the Art

- Existing interconnection networking technology for HPC systems
  - InfiniBand:
    - high bandwidth and low latency
    - Flexible topologies: Fat-tree topology for large scale systems, also support other topologies
    - Simple control: destination based routing - inflexible and inefficient
    - Performance issues when scale up
  - Proprietary technologies:
    - Less flexible topologies
    - Complex control
      - Cray Cascade (Edison):
        - » Dragonfly topology
        - » Global adaptive routing
      - IBM Bluegene:
        - » 3D and 5D torus topologies

Ethernet and SDN/Openflow will be persistent!

# InfiniBand and SDN



- InfiniBand has some SDN functionality.
- What is lacking is the per-flow resource management capability.

# Research Tasks

- Intra-domain
  1. Design OpenFlow-style SDN capability into InfiniBand
  2. Investigate the potential benefits come with the added capability
  3. Develop techniques to explore SDN capability in HPC systems
  4. Demonstrate a working SDN-enabled InfiniBand system
- Inter-domain - Demonstrate a working SDN-enabled InfiniBand system with inter-domain SDN capability
  5. Multi-domain (OpenFlow <-> OpenSM )
  6. Wide area (ex OSCARS)



# Task 1: Develop SDN-enabled InfiniBand for HPC systems.

- How to incorporate OpenFlow-style SDN capability into InfiniBand for HPC systems?
  - What is a flow for HPC application?
  - How SDN-enabled InfiniBand should work (like OpenFlow network or with modification)?
  - What changes need to be made to switches, subnet manager, subnet management agent?

# Task 2: Evaluate the potential performance benefits of SDN-enabled InfiniBand

- Assuming the best-case scenario.
- Where the SDN capability can be explored? (job scheduler, HPC application, etc)
- Active and Passive SDN
- Benefits for different types of applications (MPI, PGAS, etc)
- Simulation and modeling for small and large systems.

# Task 3: Develop techniques for exploring SDN-enabled InfiniBand

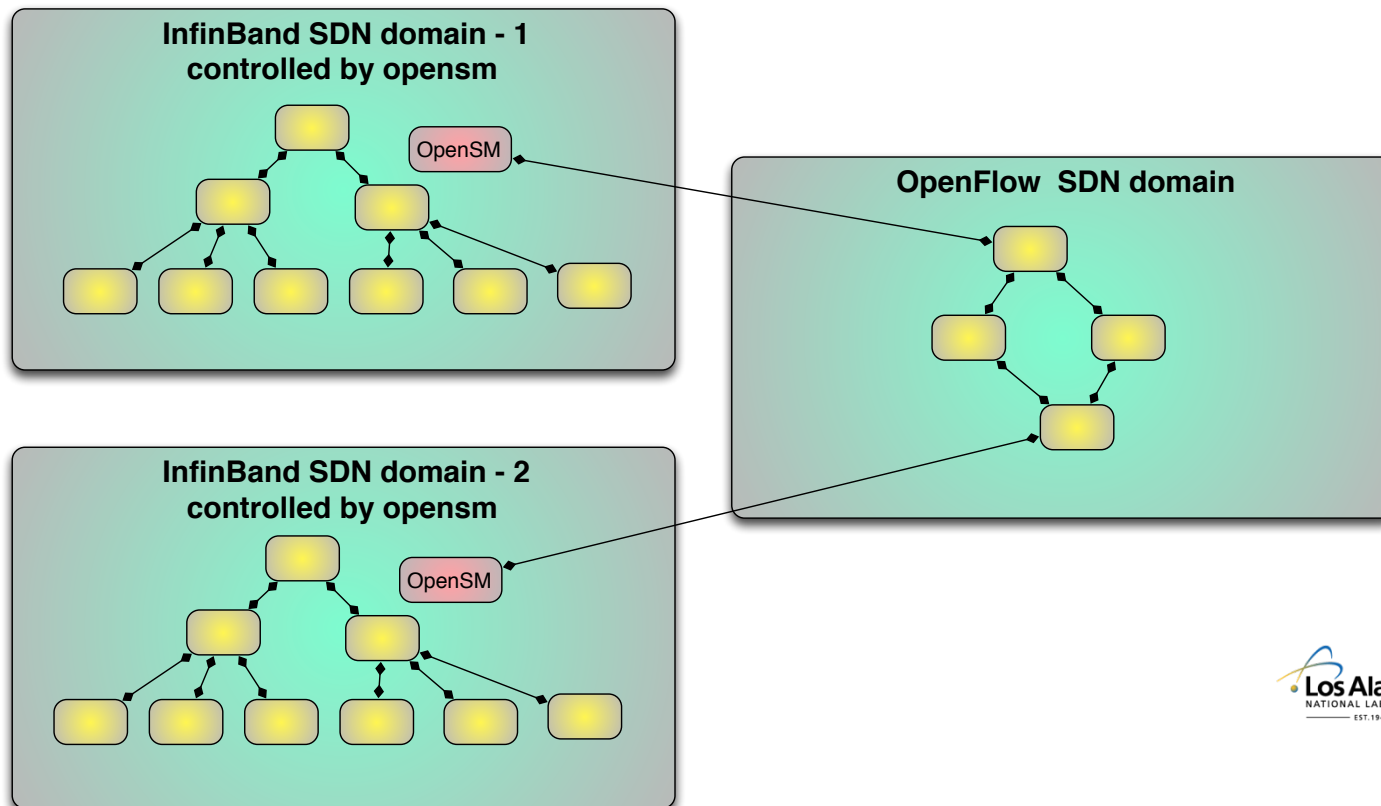
- How can the HPC system and/or HPC application effectively explore the SDN capability?
  - Techniques for exploring **passive** SDN
    - Systems level techniques – recognize flows and adjust.
  - Techniques for exploring **active** SDN
    - Scheduler/Application level techniques
  - SDN interface for HPC systems/applications
  - Resilience for SDN-enabled InfiniBand

# Task 4: Incorporate the **intra**-domain SDN functionality in OpenSM

- OpenSM is the current InfiniBand subnet management software
- Use the current multi-pathing with multiple DLIDs to emulate per-flow management functionality
- Add OpenFlow-style per-flow management capability to OpenSM
  - SDN controller functions may be incorporated into OpenSM or may be an independent entity that interacts with OpenSM
- Demonstrate OpenFlow-style SDN capability in a small scale cluster.

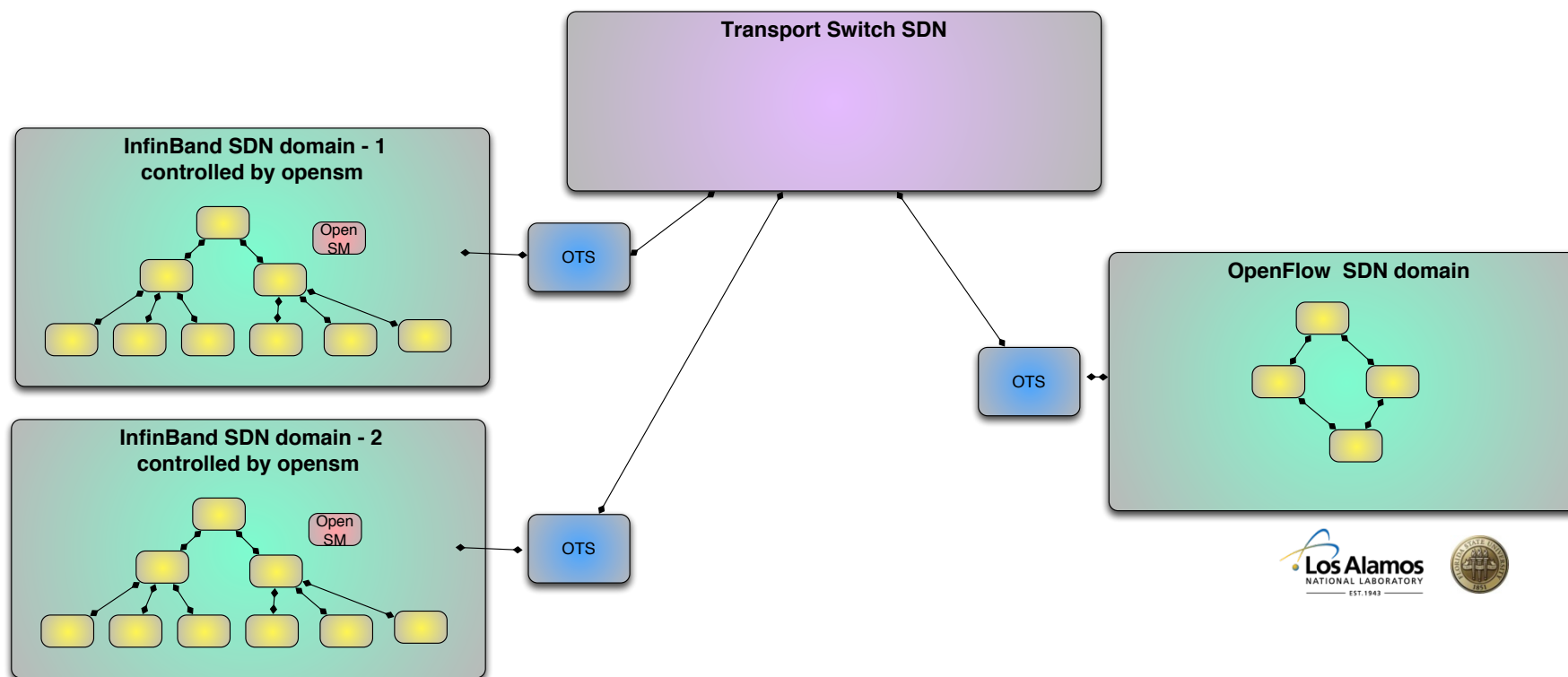
# Task 5: Augment OpenSM with distributed SDN controller functionality

- Multi-domain SDN deployment
- Add **inter**-domain functionality



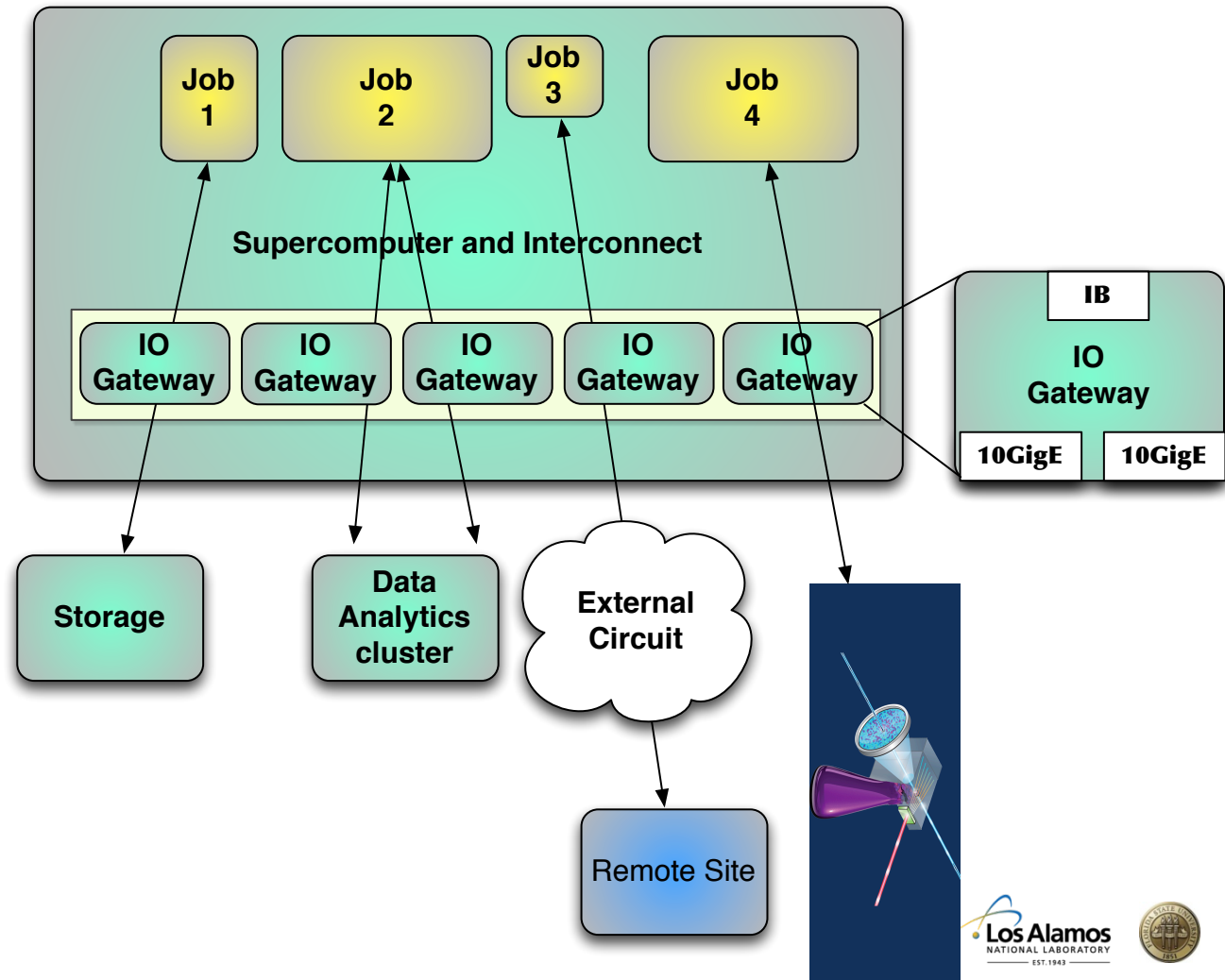
# Task 6: Map **intra**-domain SDN into inter-domain SDN frameworks

- Leverage existing inter-domain frameworks, DOE OSCARS (“orchestrators”)
- Expose internal resource and query/request external resources via OSCARS

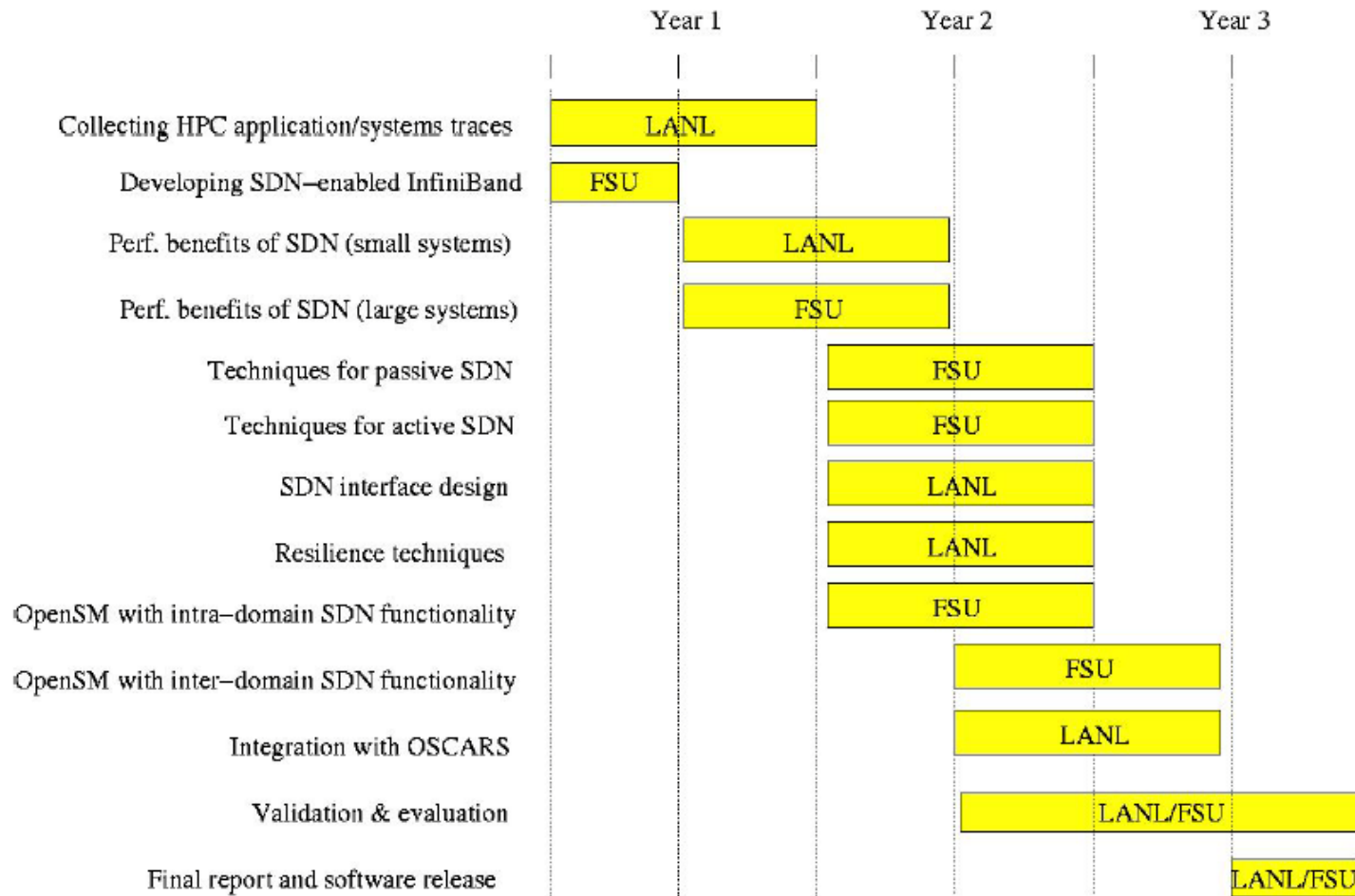


# Use cases supported with multi-domain support

- Match job requirements to external “to cluster” resources

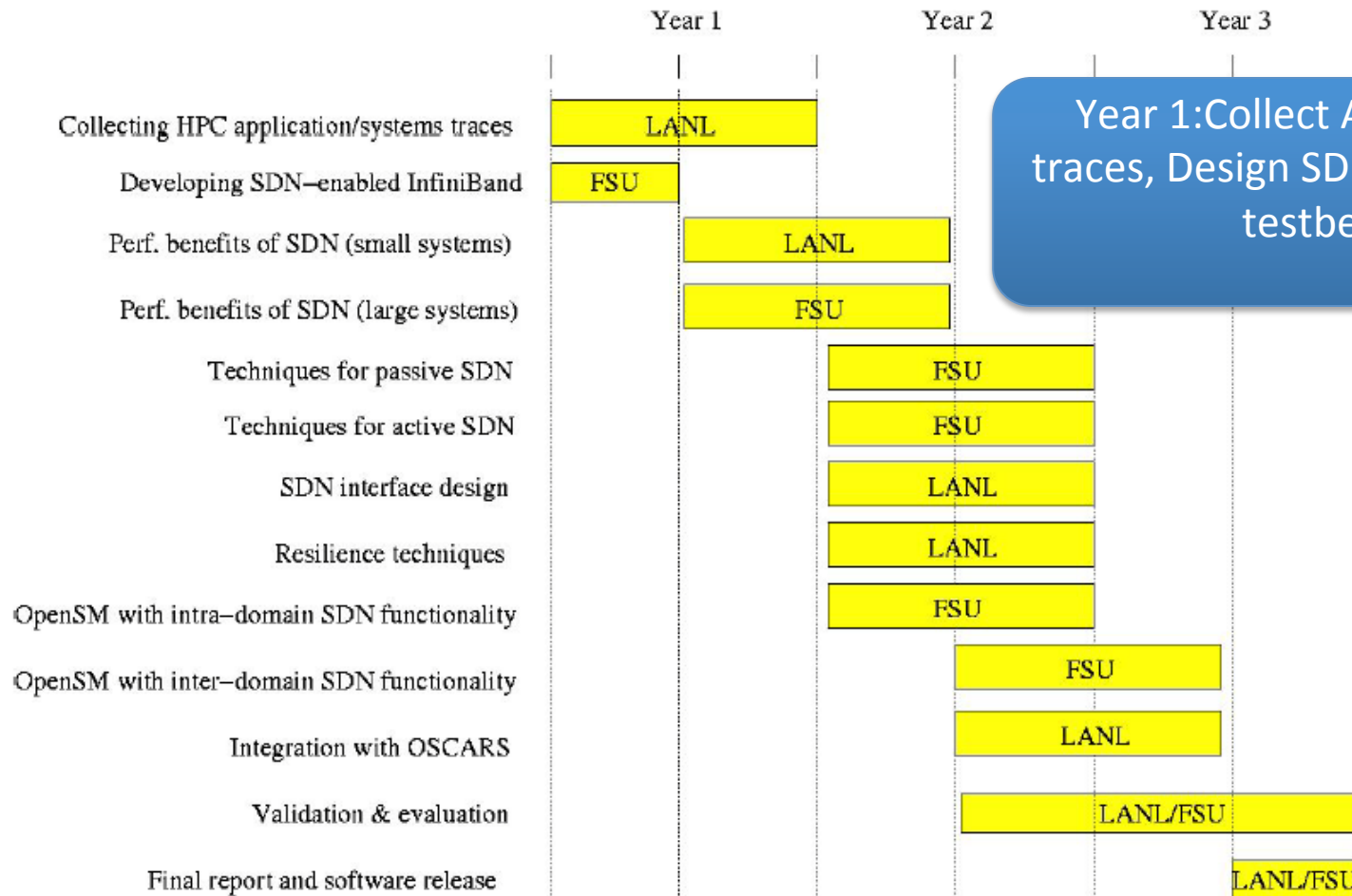


# Timeline



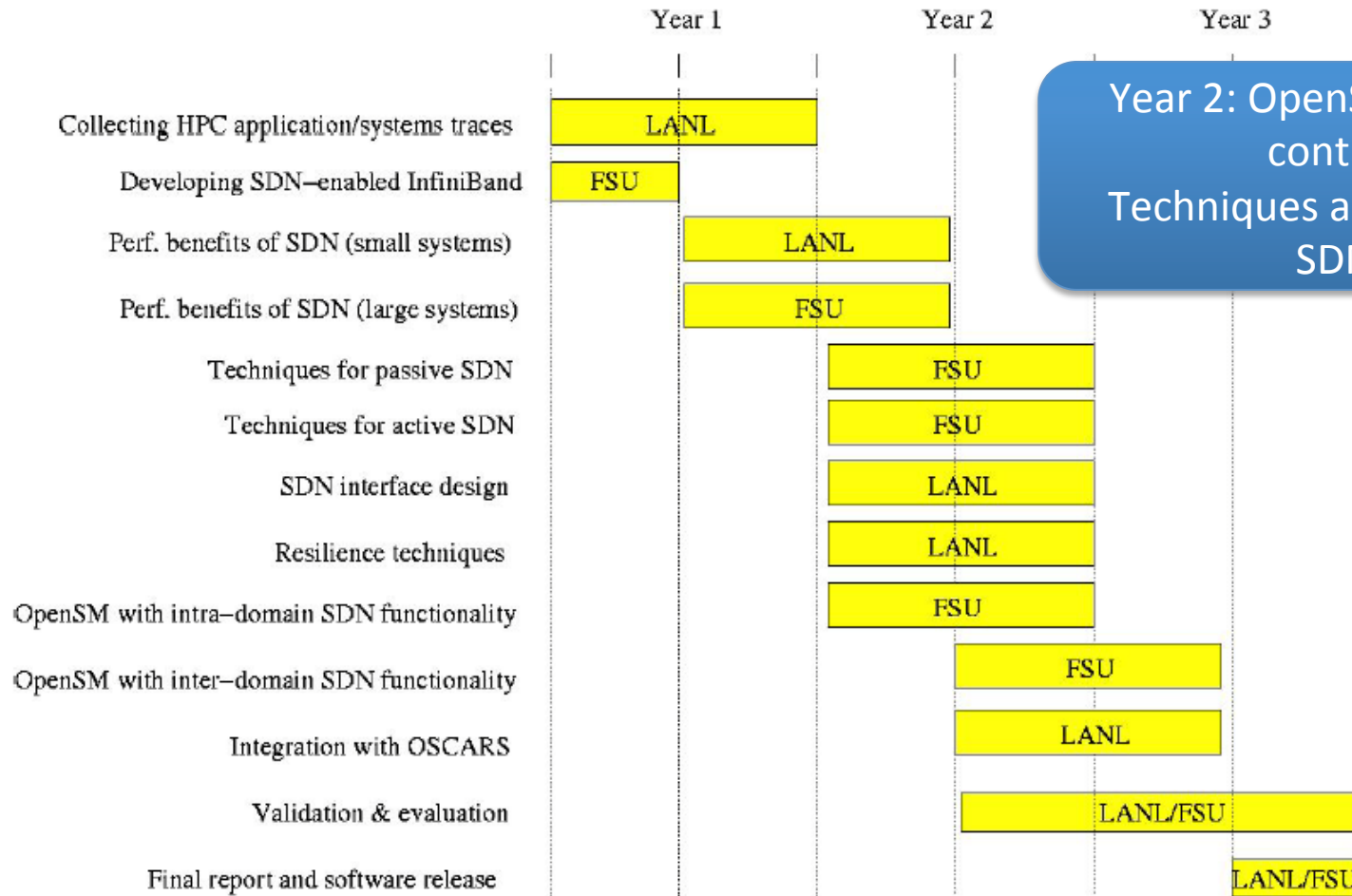


# Timeline



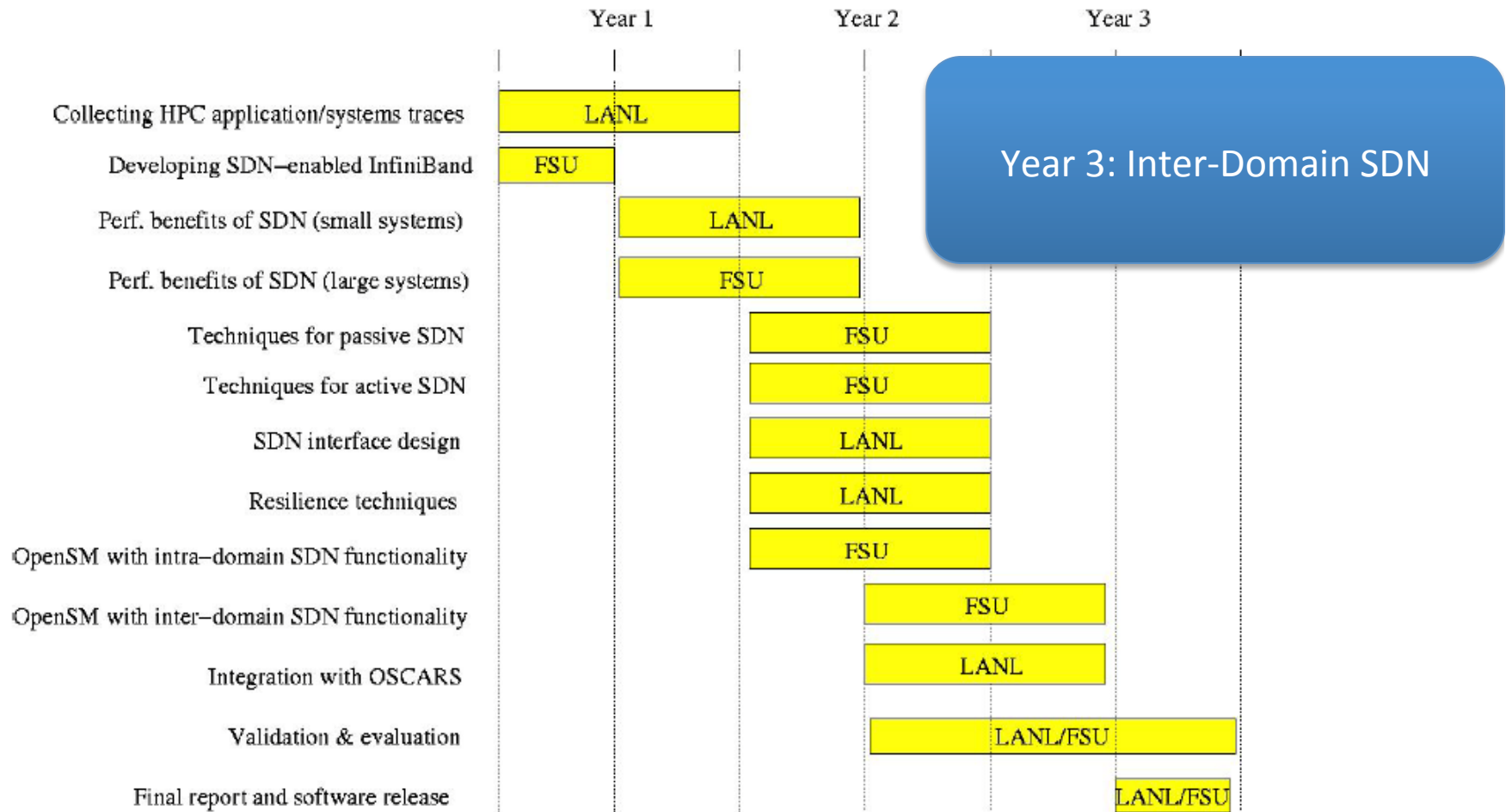
Year 1: Collect Application traces, Design SDN IB, Prepare testbed

# Timeline



Year 2: OpenSM/Openflow controller  
Techniques and Benefits of SDN IB

# Timeline



# SDN-enabled InfiniBand for HPC: a preliminary design

- Objective: Add per-flow resource management into InfiniBand
  - Pre-establishment of flow table entries
    - Flow table in addition to forwarding table
    - Flow table entries are pre-established at either job launch time or during job execution before the communication starts
      - HPC applications have phase behavior
    - Fall back to forwarding table in case of flow table miss.
      - forwarding table remains also for the network initialization.

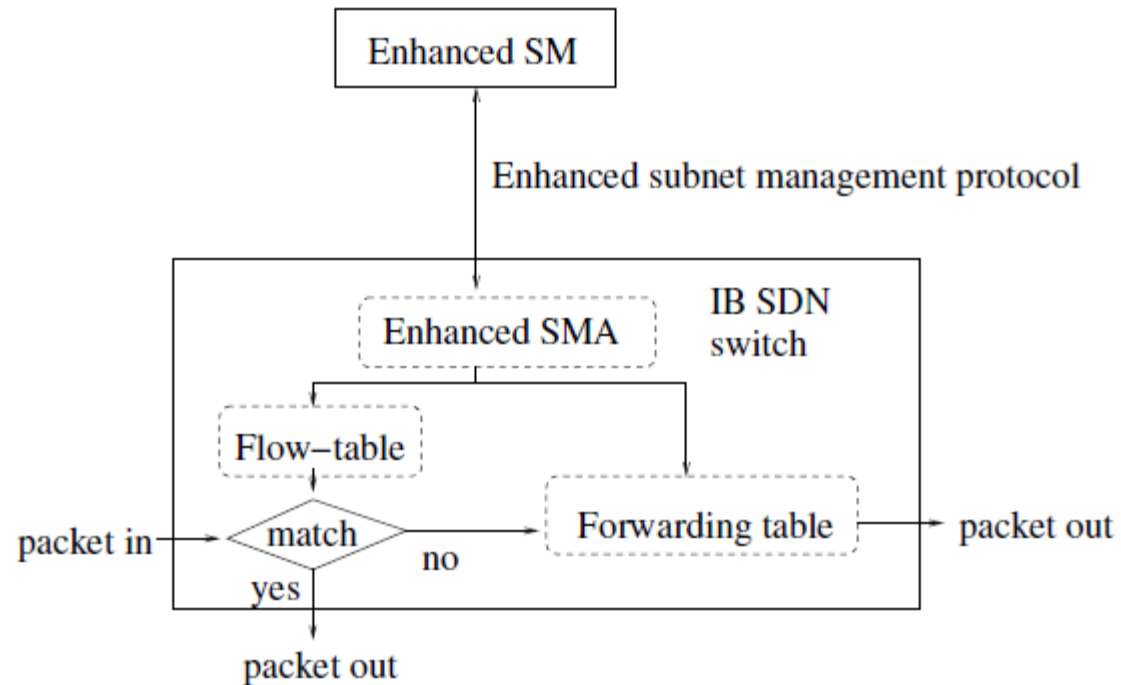
# SDN-enabled InfiniBand for HPC: a preliminary design

## – The flow concept

- Use header fields in the existing InfiniBand packets
- Flow identifier depending on per-flow functionality needed and flow table constraints.
  - DLID: allows for per application destination based routing
  - DLID+SLID: allows for per application source/destination routing
  - DLID+SLID+SL: allows for per application source/destination routing with multiple levels of service quality
  - DLID+SLID+SL+DestQP: per application multi-path routing based on flows
  - DLID+SLID+SL+DestQP+PSN: packets for the same message follow different routes

# SDN-enabled InfiniBand for HPC: a preliminary design

## – SDN-enabled InfiniBand Switch



# SDN-enabled InfiniBand for HPC: preliminary design

- OpenFlow Control packet Modification:
  - Add a new subnet management class for OpenFlow functionality (ManagementClass 0x09)
  - The data field follows OpenFlow packet format.

Field	Bits	Value	Used	Comment
BaseVersion	8	1	Yes	Required
MgmtClass	8	0x09	Yes	Vendor specific value (OpenFlow)
ClassVersion	8	1	Yes	Required
R	1	1 0	Yes	Depends on direction
Method	7	0	No	
Status	16	0	No	
Class Specific	16	0	No	
TransactionID	64	#	Yes	Generated from InfiniBand header data <sup>1</sup>
AttributeID	16	0	No	
Reserved	16	0	No	Reserved
AttributeModifier	32	0	No	
Data	??	ofp_flow_mod	Yes	Depends on direction

# SDN-enabled InfiniBand for HPC: a preliminary design

## – Addition to OpenSM

- Implement subset of OpenFlow SDN controller functions
  - Maintain the global status of flow tables in the network
  - Interacting with applications, compute flow table entries, and set-up forwarding table entries
- Two implementation choices:
  - Integrated within OpenSM
  - An independent controller outside OpenSM



# SDN-enabled InfiniBand for HPC: a preliminary design

## – Addition to OpenSM

- Implement subset of OpenFlow SDN controller functions
  - Maintain the global status of flow tables in the network
  - Interacting with applications, compute flow table entries, and set-up forwarding table entries
- Two implementation choices:
  - Integrated within OpenSM
  - An independent controller outside OpenSM

# SDN-enabled InfiniBand for HPC

## Summary:

- Software deliverables
  - OpenSM + OpenFlow interoperability
- Design of IB + SDN
- Demonstrate capability of IB + SDN
- Evaluation of possible performance improvements

Baseline: IB independently managed from external networks.

New functionality: IB and SDN managed, in concert, to allow “*smarter* IB” for scientific applications and workflows

Synergy: *Expose* IB resources/interact with SENSE, FLOWS, “orchestrators”

\* Try to influence future Infiniband hardware

# Questions

# SDN-enabled InfiniBand for HPC: a preliminary design

## – Addition to OpenSM

- Implement subset of OpenFlow SDN controller functions
  - Maintain the global status of flow tables in the network
  - Interacting with applications, compute flow table entries, and set-up forwarding table entries
- Two implementation choices:
  - Integrated within OpenSM
  - An independent controller outside OpenSM