SAND2009-4678P

RedSky IB Torus Infrastructure

John Naegle
Matthew Bohnsack
Marcus Epperson
Jim Monk
Jim Schutt

New Features in RedSky

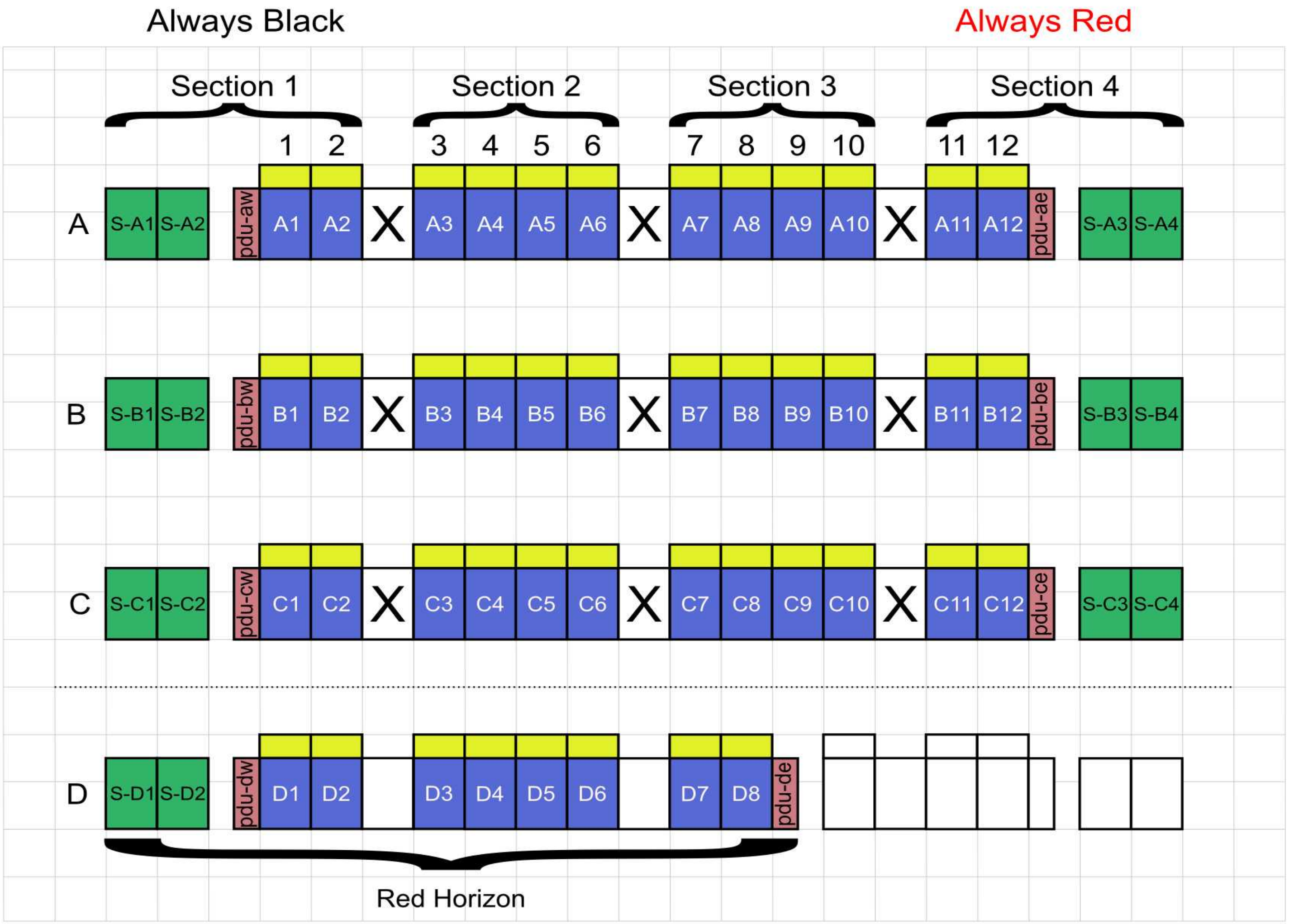
- Gas cooled doors: extremely efficient
- Efficient 240 volt rather than 208 power units
- Nehalem processors:
  - Initial 2x improvement in user codes over AMD
- Unified data fabric using Torus
  - QDR IB is the only data network
  - No external Ethernet or IB switches
    - Significant cost and power savings
  - Potential for reasonable Red/Black switching

- RedSky is a high value, green machine!

Benefits of the Torus Architecture

- 12x QDR paths in each dimension maintains reasonable bisection bandwith/FLOP ratio
- Regular wiring enables Red/Black switching
- Scales linearly
- Works well for localized communication, particularly in capacity environment
- Potential for QOS
- Save cost, power, and cooling of external fat-tree IB switches
- Save cost, power, cooling, and cabling of high-speed Ethernet infrastructure

Physical layout of RedSky



Always Black

Always Red

Section 1 Section 2 Section 3 Section 4

1 2 3 4 5 6 7 8 9 10 11 12

A S-A1 S-A2 pdu-aw A1 A2 X A3 A4 A5 A6 X A7 A8 A9 A10 X A11 A12 pdu-ae S-A3 S-A4

B S-B1 S-B2 pdu-bw B1 B2 X B3 B4 B5 B6 X B7 B8 B9 B10 X B11 B12 pdu-be S-B3 S-B4

C S-C1 S-C2 pdu-cw C1 C2 X C3 C4 C5 C6 X C7 C8 C9 C10 X C11 C12 pdu-ce S-C3 S-C4

D S-D1 S-D2 pdu-dw D1 D2 D3 D4 D5 D6 D7 D8 pdu-de

Red Horizon

□ 2' x 2' raised floor tile

🟨🟦 C48 compute rack with glacier door facing north

🟩 Disk storage and support intrastructure rack

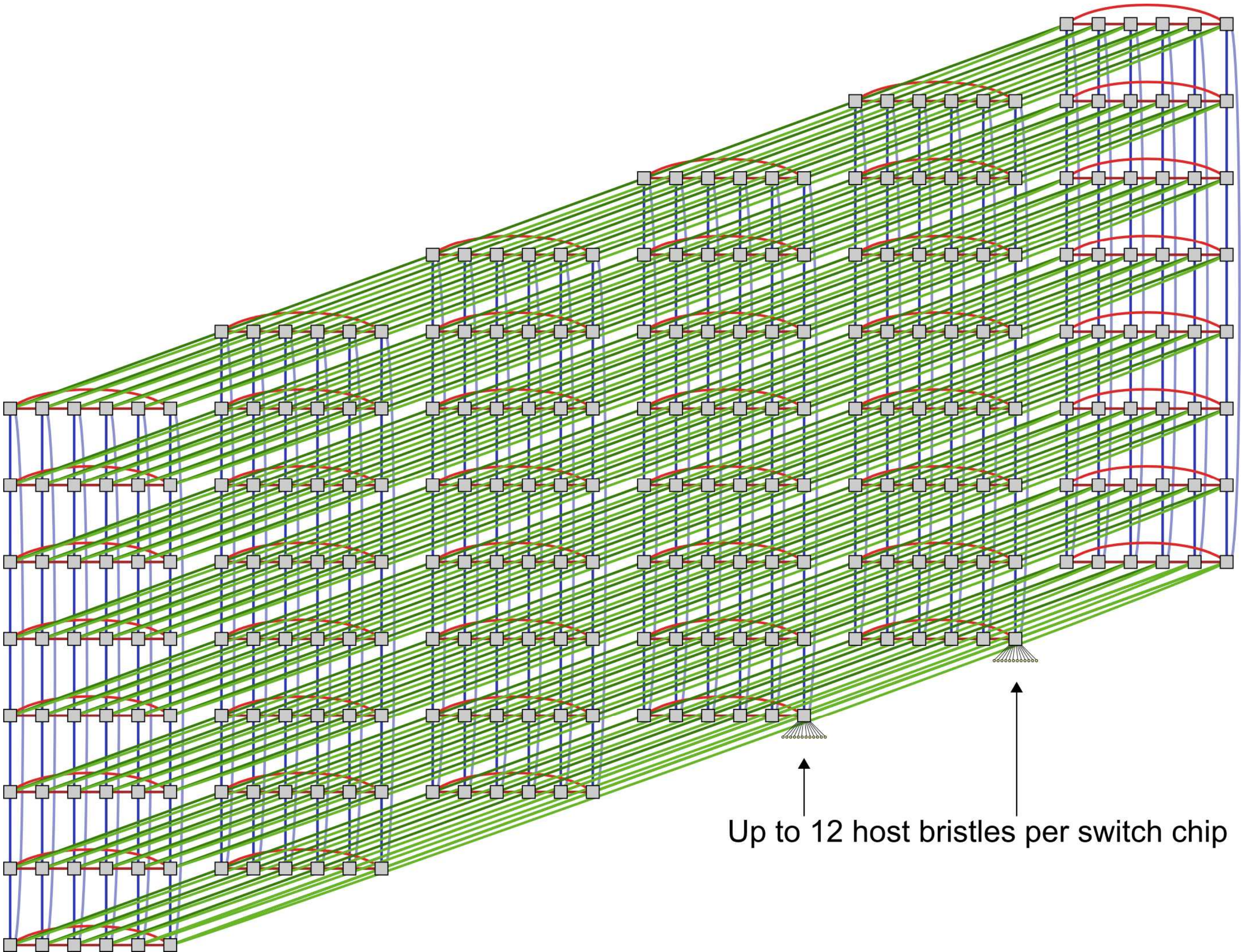
☒ Red/Black switching disconnect rack

□ Empty rack or planned expansion

🟥 APC PDU

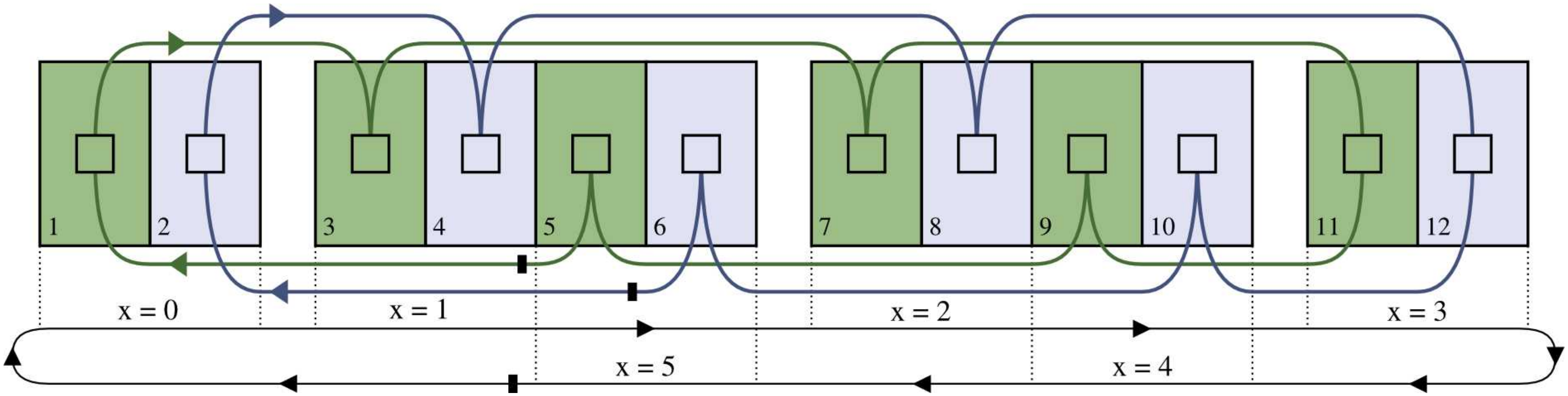
N
W E
S

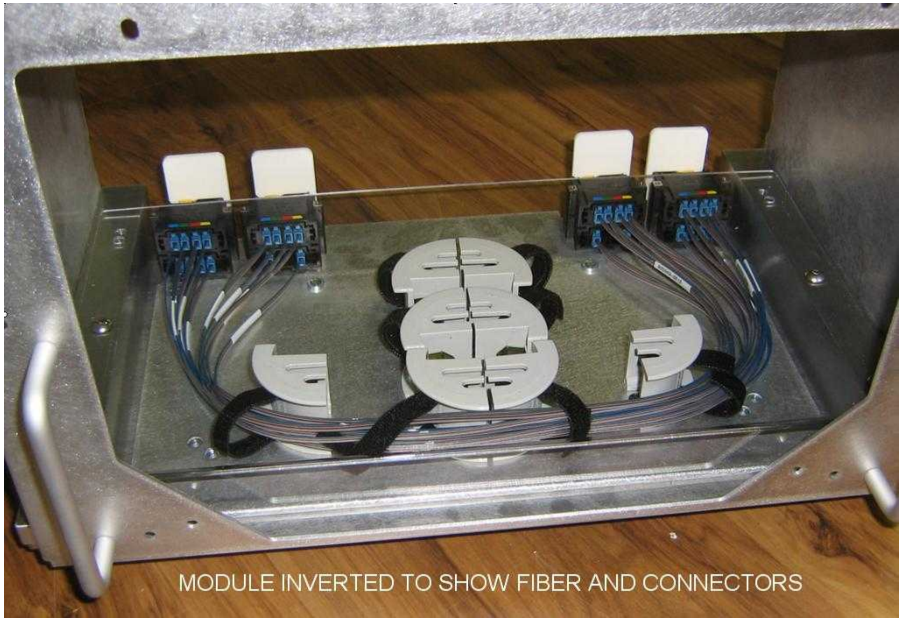
Logical Layout of RedSky

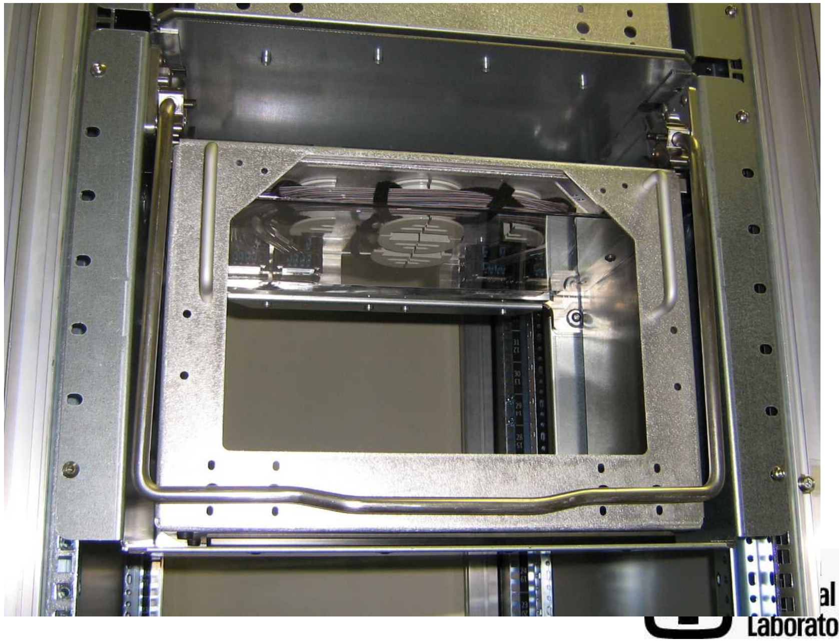


Up to 12 host bristles per switch chip

Red/Black Switching



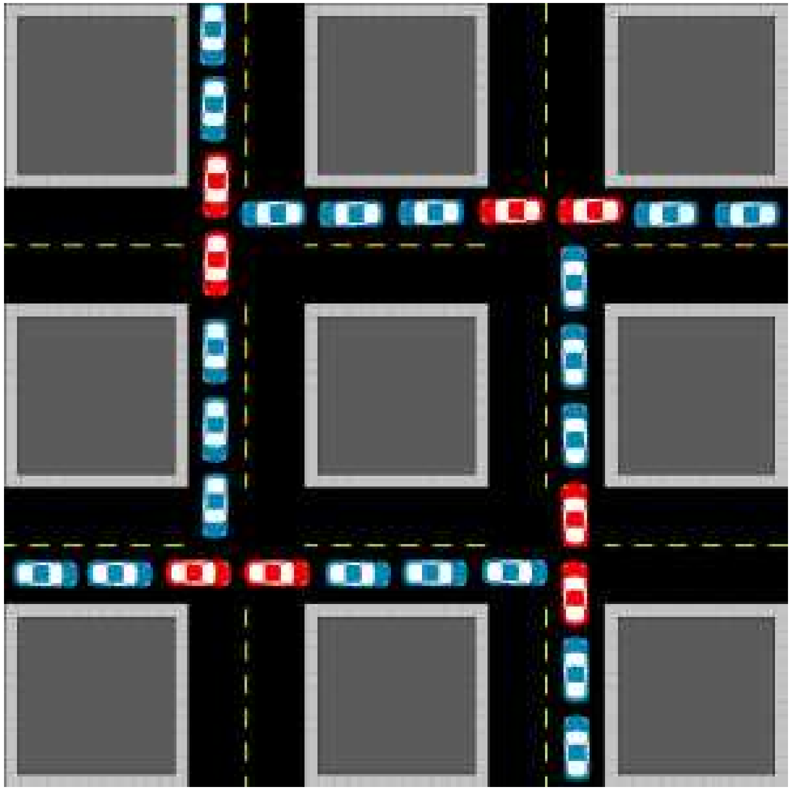

MODULE INVERTED TO SHOW FIBER AND CONNECTORS



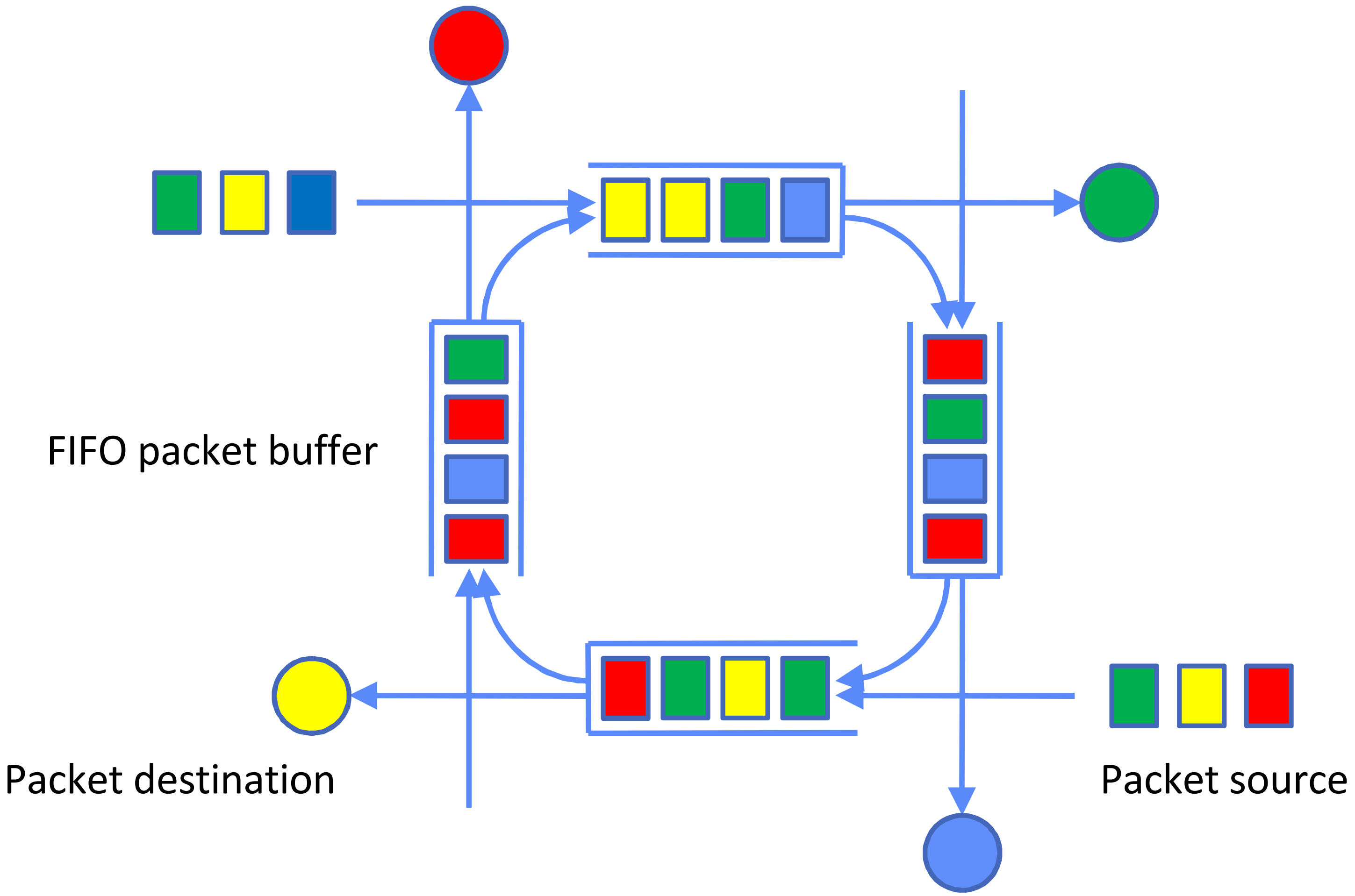
Difficulties of IB Torus

- Torus Susceptible to deadlock routing
- IB NOT designed for deadlock free routing
  - Not capable of turn based methods
  - Must use constant SL determine at source
  - Must share SL function with QOS implementation
  - Limited by SL to VL mapping and fixed sizes
  - Must use Path Record queries for connection setup
  - Resiliency to switch or link failures very difficult

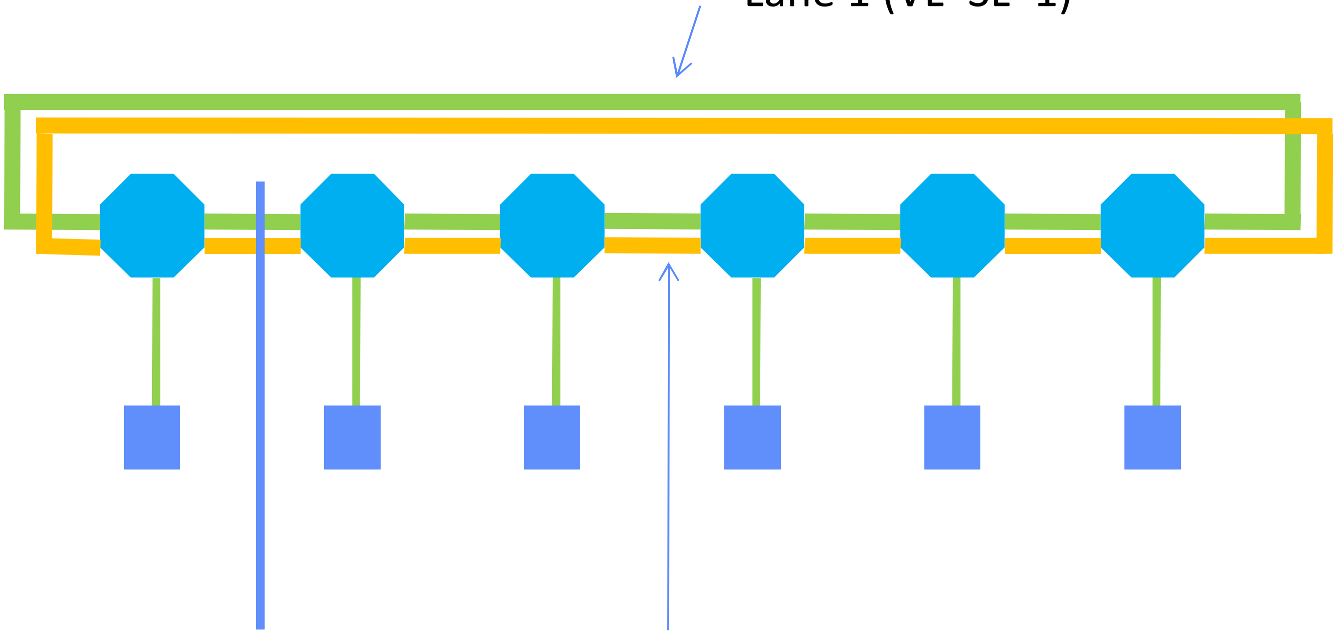
Deadlock is Gridlock for Bits



Simple Deadlock
Credit Loop



Deadlock Avoidance



Lane 1 (VL=SL=1)

Lane 2 (VL=SL=2)

Dateline

Route Crosses -> SL=2

Route Not Cross -> SL=1

Deadlock Free Routing

- LASH
  - Algorithm to map each route and add SLs when needed to avoid loops
  - Modified existing algorithm to utilize basic Dimension Order Routing to minimize SLs
  - Requires non-existent Path Record Update implementation for resilience to failures
- Jim Schutt algorithm
  - Novel technique to use source port to determine illegal turns and utilize secondary routing
  - Heuristically demonstrated, no mathematical proof
  - Implementation currently being debugged using standard OFED tools
  - Does NOT require Path Record Updates

Other Issues

- Requires applications to use Path Record Queries to determine launching SL
  - Inherently difficult to scale, investigating options such as static tables, caching, or distributed SM
  - OpenMPI RDMA-CM implementation broke
    - Patches to several bugs already submitted for 1.3.3
      - Assert, qpair release, retries, CTH bug
- QOS also uses SLs, combined solution limits number of QOS levels to two
- Many of the basic IB management tools did not work with SL != 0 and needed to be fixed
- MVAPICH2 not working with RDMA-CM
- Demonstrating MESH as the fall-back