Cloud Computing to Support Experiment Online Computing

Fall 2020 HEPiX
October 15, 2020
Scientific Data and Computing Center (SDCC)
Shigeki Misawa

BROOKHAVEN NATIONAL LABORATORY
U.S. DEPARTMENT OF ENERGY
BROOKHAVEN SCIENCE ASSOCIATES

Brookhaven National Laboratory

US Department of Energy National Laboratory

- One of ten Office of Science laboratories
- Multi-disciplinary science laboratory

“Big Data” experimental facilities at BNL

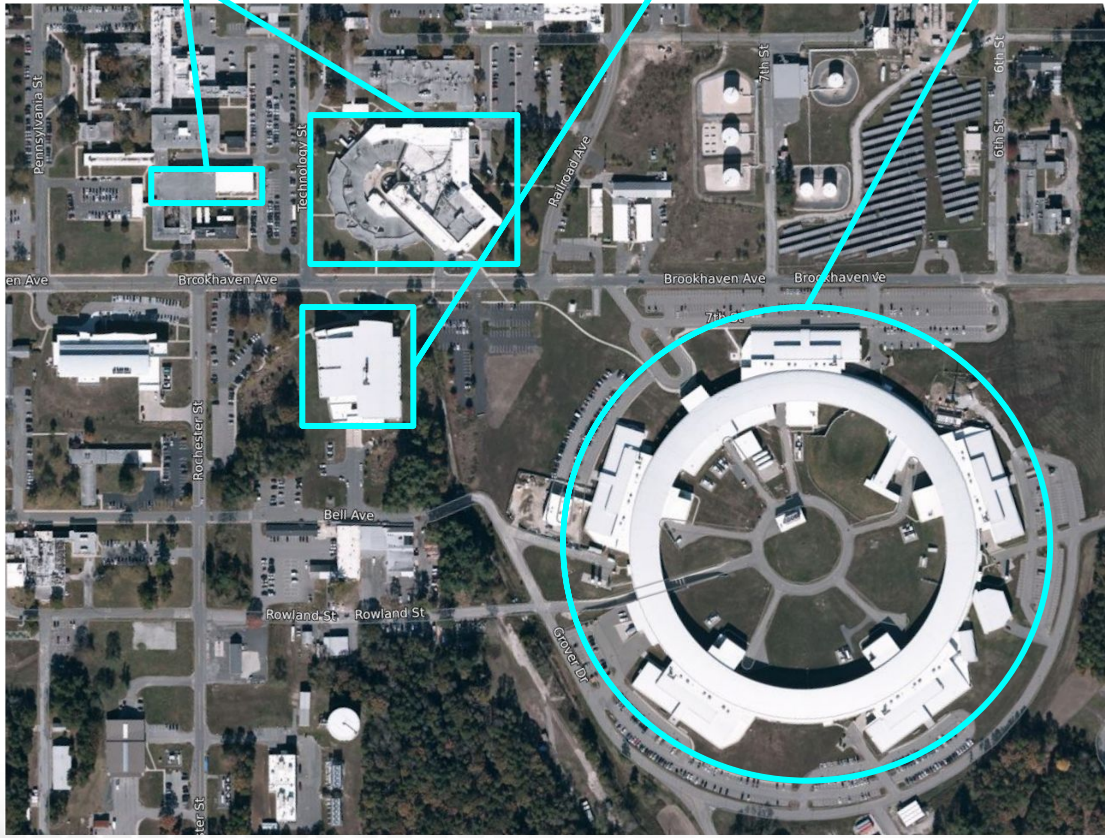
- Relativistic Heavy Ion Collider (RHIC)
- National Synchrotron Light Source II (NSLS-II)
- Center for Functional Nanomaterials (CFN)
- *Future* Electron Ion Collider (EIC)

Background

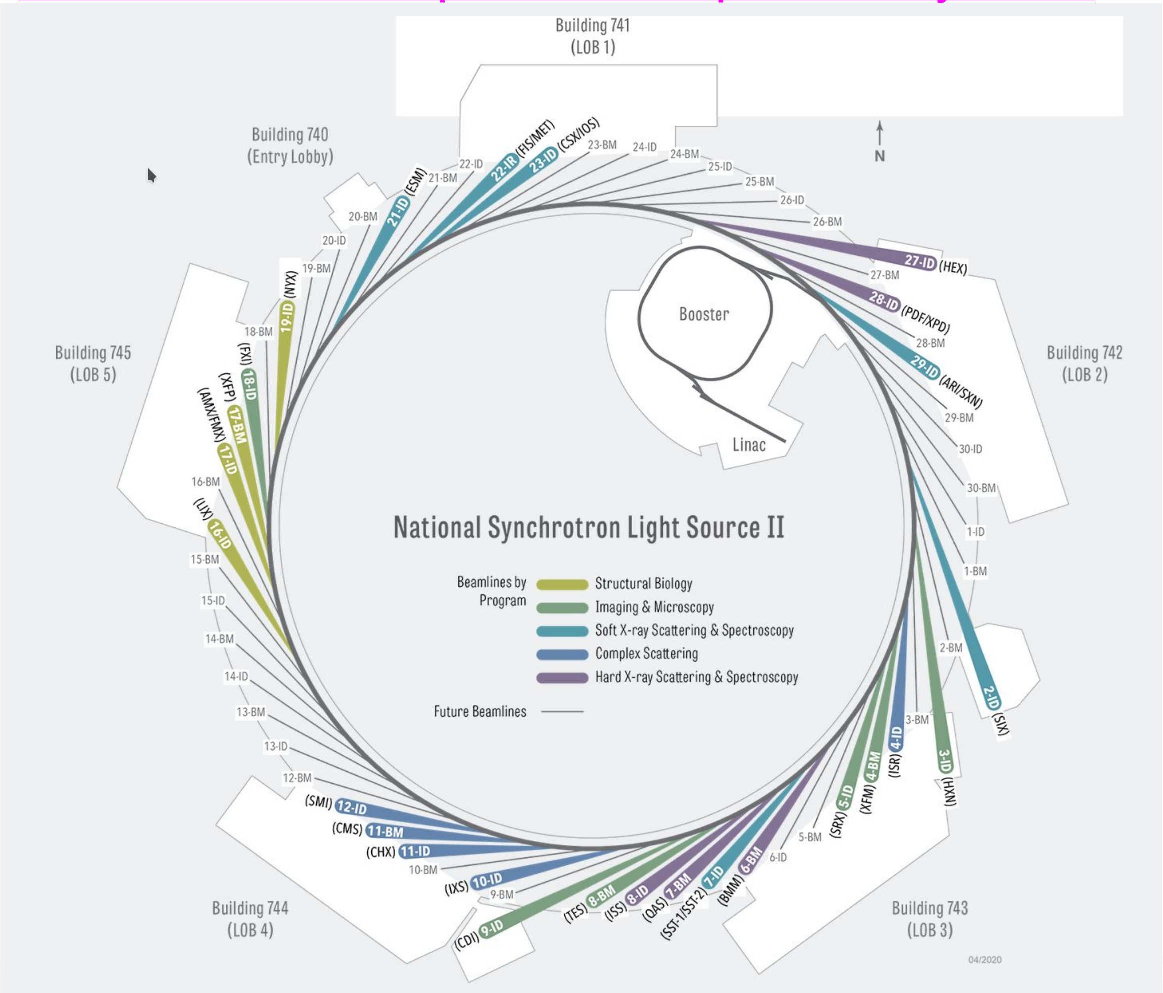
- “Democratization” of “Big Data” experiments
  - Proliferation of unique “big data” instruments at BNL
    - Data rates up to 50 GB/sec (note GigaBYTES, not Bits)
  - Geographically dispersed on campus
  - Each instrument serially hosts multiple experiments per year
  - Instruments may not be in use 24x7
  - Researchers require fast, on line analysis for “on the fly” adjustments to experiments
  - Compute/storage requirements may vary between instruments and between experiments using a single instrument
  - Budget and infrastructure limitations prevent deployment of necessary compute at the instrument
  - **<u>Requires support from the Data Center</u>**

Instruments at Brookhaven

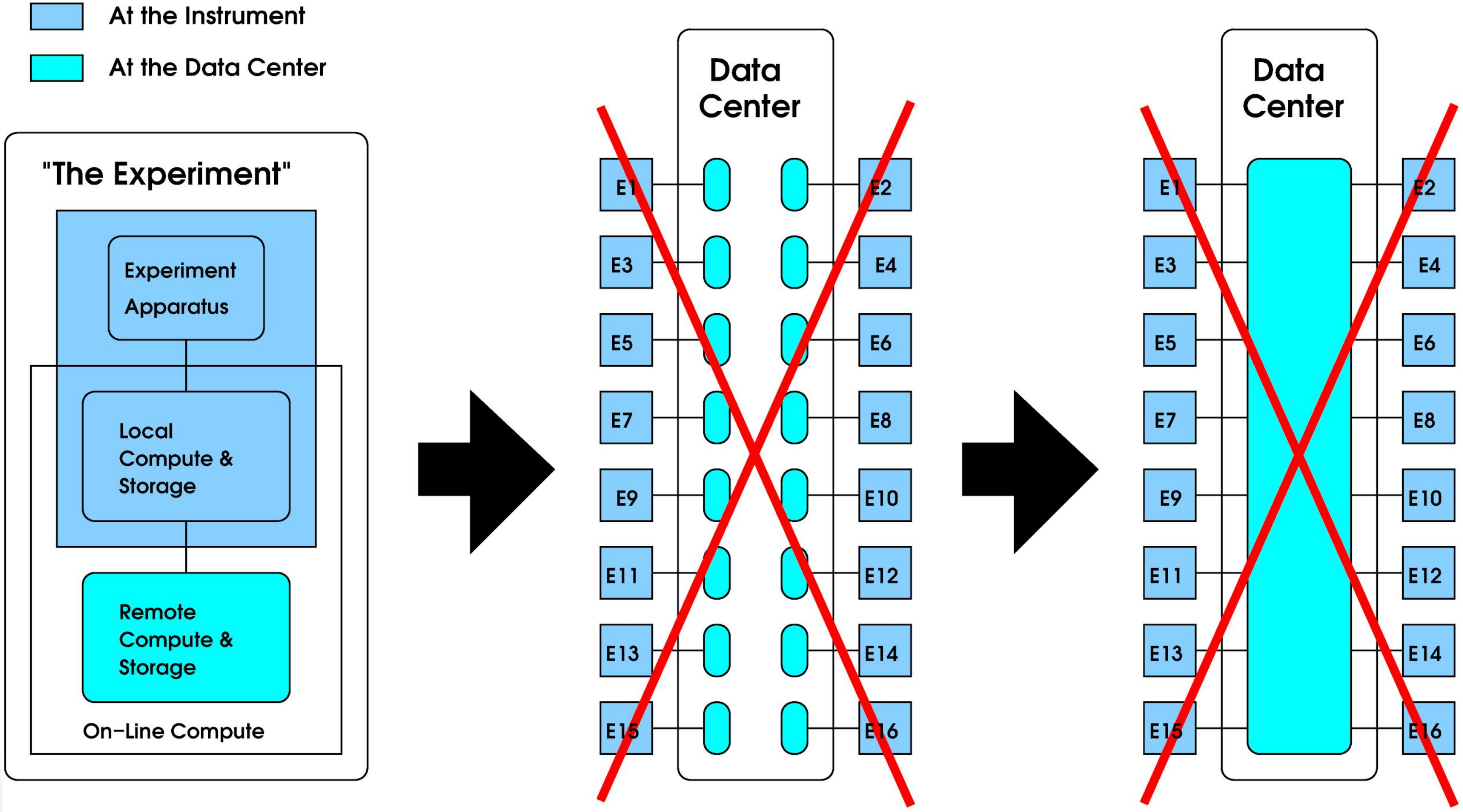
Data Center CFN NSLS-II



28 beamlines in operation, expect 40 by 2030



Supporting New Big Data at Scale



Option A

Option B

Problems with Scaling Support

- Option A
  - Underutilization of compute resources as experiments don’t run 24x7
  - Difficult to move resources between experiments
  - Difficult to retask resources for off line or data center use.
  - Not cost efficient
- Option B
  - Problematic security
    - Systems at each experiment visible to other experiments.
    - Each experiment potentially visible from interactive compute nodes in the data center
      - Note that Slurm allows users to log into batch nodes running their jobs.
    - DTN between experiment and data center fixes security, but adds complexity for researchers
    - Firewalls fixes security, but may impact performance and cost.

Notes on Security

- Security is a major goal for the virtual online computing concept
  - If security were a non-issue, problem would be solved with a single, open network fabric providing all to all connectivity (Option B in previous slide)
- Core Security Principles
  - Each experiment (i.e., equipment at the instrument) should be protected from other experiments and the data center by default
  - Should be possible to grant/revoke trust between instrument sites as required by the experiment
  - Data center limits trust of equipment at experiment site
  - Experiment site trusts data center staff but not users at the data center

Goals for Virtual On-Line Computing

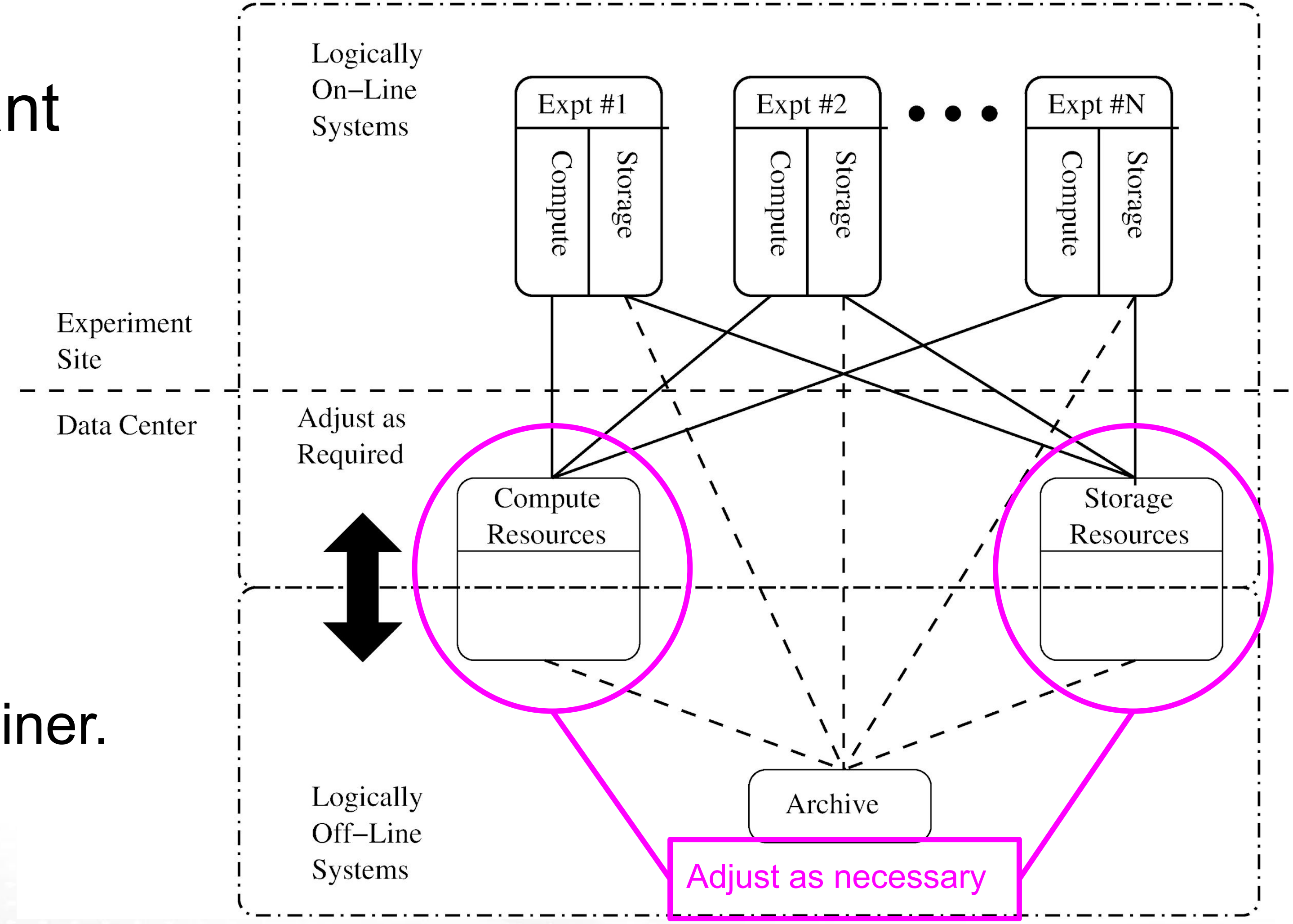
- Location independence of compute (and storage)
  - Make remote (data center) resources available for use “at” the experiment
- “Re-task” compute and storage as required
  - Move resources between different experiments as needed
  - Move resources between on-line and traditional off-line/data center use
- Achieve high utilization of resources to reduce costs by sharing equipment among the experiments
- Enable high bandwidth, low latency movement of data
  - Minimize or eliminate “store (to disk) and forward” in the data stream
  - Enable direct streaming from instrument or DAQ to data center based resources (No DTN)
- Maintain security of the instrument, DAQ, and data center.

Parallels with Commercial Cloud Services

- Goals largely the same for cloud providers (Amazon, Google, Microsoft) hosting private data center extensions for multiple customers
  - Secure connectivity from cloud resources into the heart of a client's enterprise data center
  - Dynamic allocation of resource to meet customer demands
  - Multi-tenancy to allow for full utilization of resources
    - Isolate tenants from each other
- Cloud provider solution is data center virtualization
  - Server virtualization or containerization
  - Network virtualization/dynamic reconfiguration

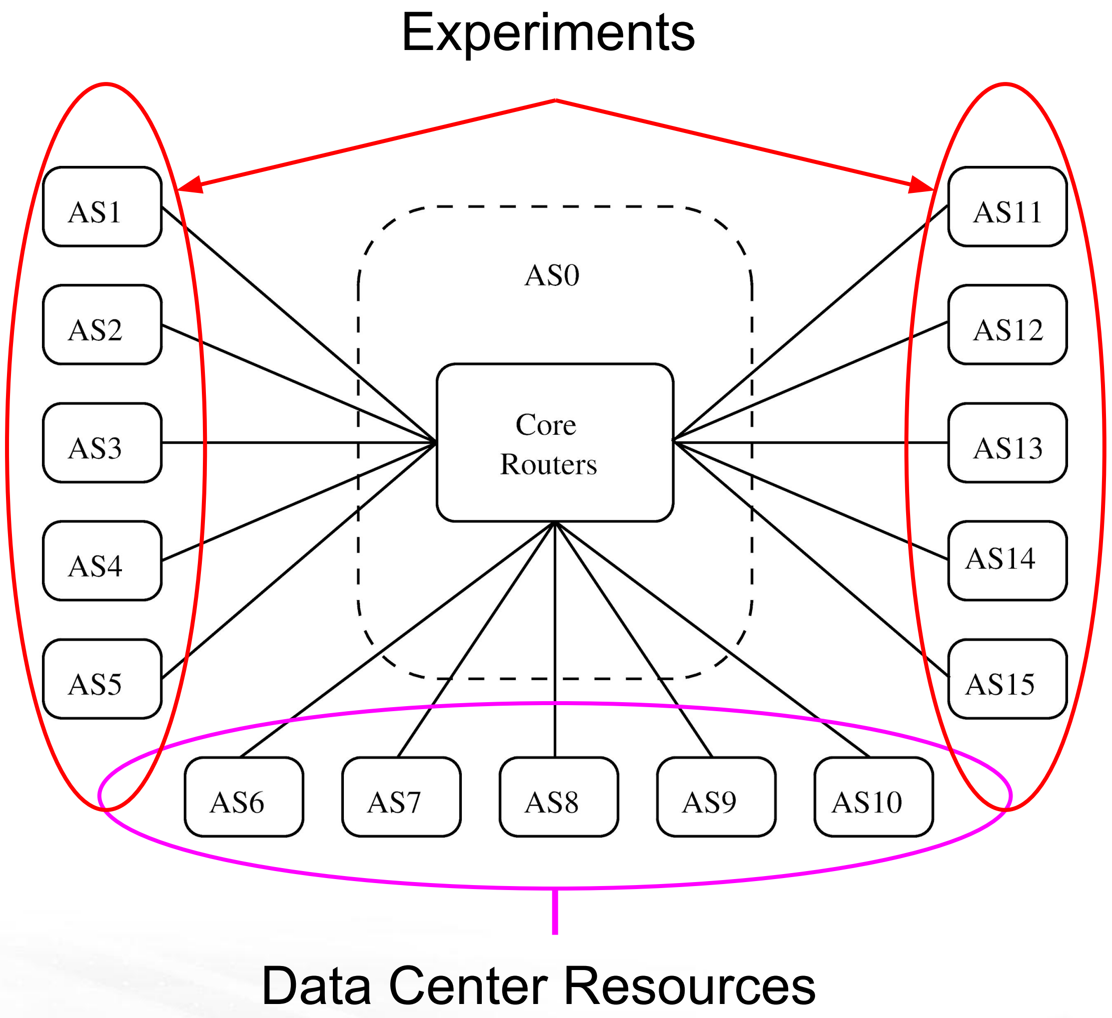
Virtual On-Line Computing Architecture

- Build a private multi-tenant cloud in the data center
- Keys Components
  - Network configuration
    - Control connectivity
  - Node provisioning
    - Allocate resources
  - Node configuration
    - bare metal, VM, container.



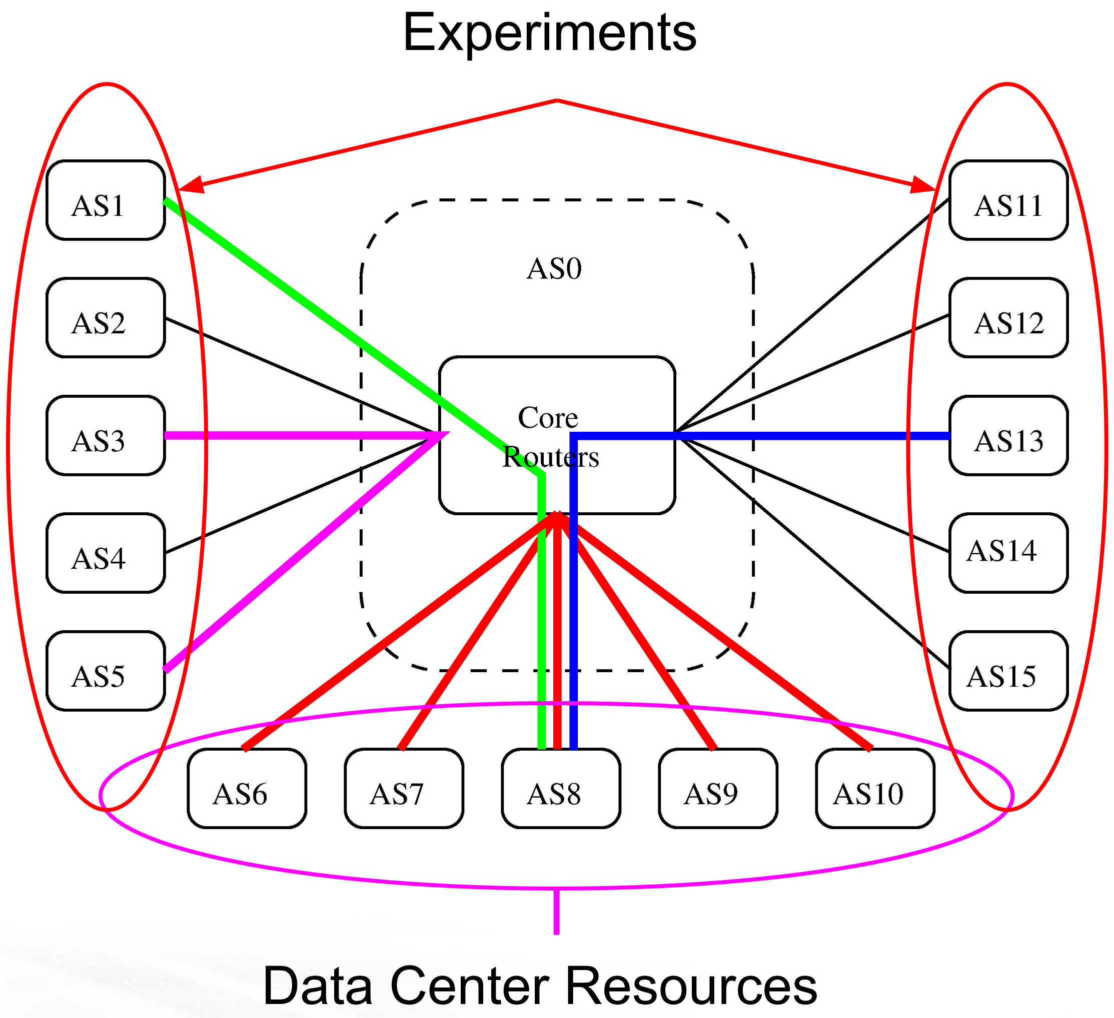
Network Foundation

- Dedicated routed network fabric
- Systems grouped into "Autonomous Systems" (ASn)
  - Data center resources partitioned into multiple ASn
  - Each experiment site is an ASn
  - Routes propagated by Border Gateway Protocol (BGP)
- Core Routers gate connectivity
- No need for DTNs/Firewalls
- In use at BNL since late 2015



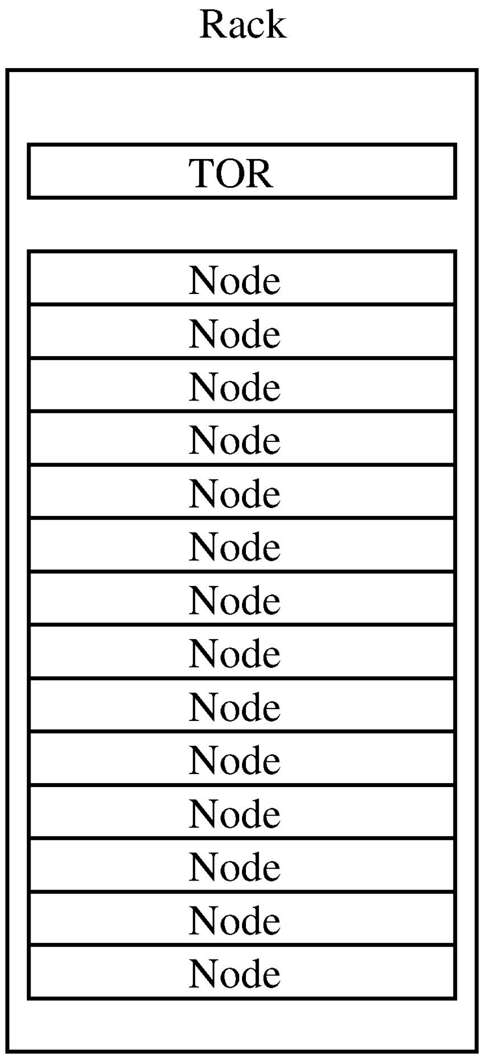
Network Fabric in Action

- Selected BGP route advertisements denoted by colored lines
- AS1 and AS13 can access resources in AS8
- AS1 and AS13 are not connected
- AS3 and AS5 are connected
- AS6, AS7, AS8, AS9, and AS10 are interconnected, but except for AS8, can't access AS1 or AS13



Compute Foundation

- Compute Resource Pool in the data center
  - Racks of compute nodes
  - Top of Rack (TOR) switch
  - Spine and Leaf Topology
  - BGP between Spine and Leaf
  - Each rack is an ASN (by default)
- Allocated resources can be
  - Bare Metal
  - Virtual Machine
  - Container
- Nodes/Racks
  - Can support "multi-tenancy" if necessary



Creating the virtual on-line computing system

- Building blocks in place for a virtual on-line computing system
- System can be instantiated in multiple ways.
- Open questions for different implementations
  - Is there institutional “buy in” ?
  - What is the best fit based on experiment requirements ?
  - What is the effort involved in building and deploying ?
  - Do all the necessary hardware capabilities exist ?
  - Does all the necessary software exist to achieve the desired level of automation ?
  - How mature (stability and longevity) are the components ?
  - Will it meet requirements and budget limitations ?

Path Forward

- Compute resource management
  - Expect leading solution to be container based (e.g. K8s)
    - Lighter weight than VMs, more flexible than bare metal
  - More complete solutions for VMs and containers compared to bare metal
  - Expect lower system overhead with containers compared to VMs
  - OpenStack, OpenNebula, Kubernetes and other cloud platforms likely to cover the compute aspects of the virtual online computing system
- Network resource management/automation
  - Cloud platforms may be capable of managing the network
  - Two domains of concern
    - Data center internal network - Compute node to core routers
    - Data center external network - Core routers to experiment site

Internal Network

- Internal Network
  - Cloud platform software networking components likely to provide most of the capabilities required
    - Configuration of container network configuration
    - Container host node configuration
  - Broadest software support with VXLAN, but not optimal solution.
  - Non overlay alternatives to VXLAN preferred
    - Possible, but dependent on installed hardware and software.
    - Deployed hardware might not have necessary capability (VRFs)
    - Without software to make fast, programmatic router changes, slower manual intervention would be necessary
  - Additional research required.

Choices for Internal Network

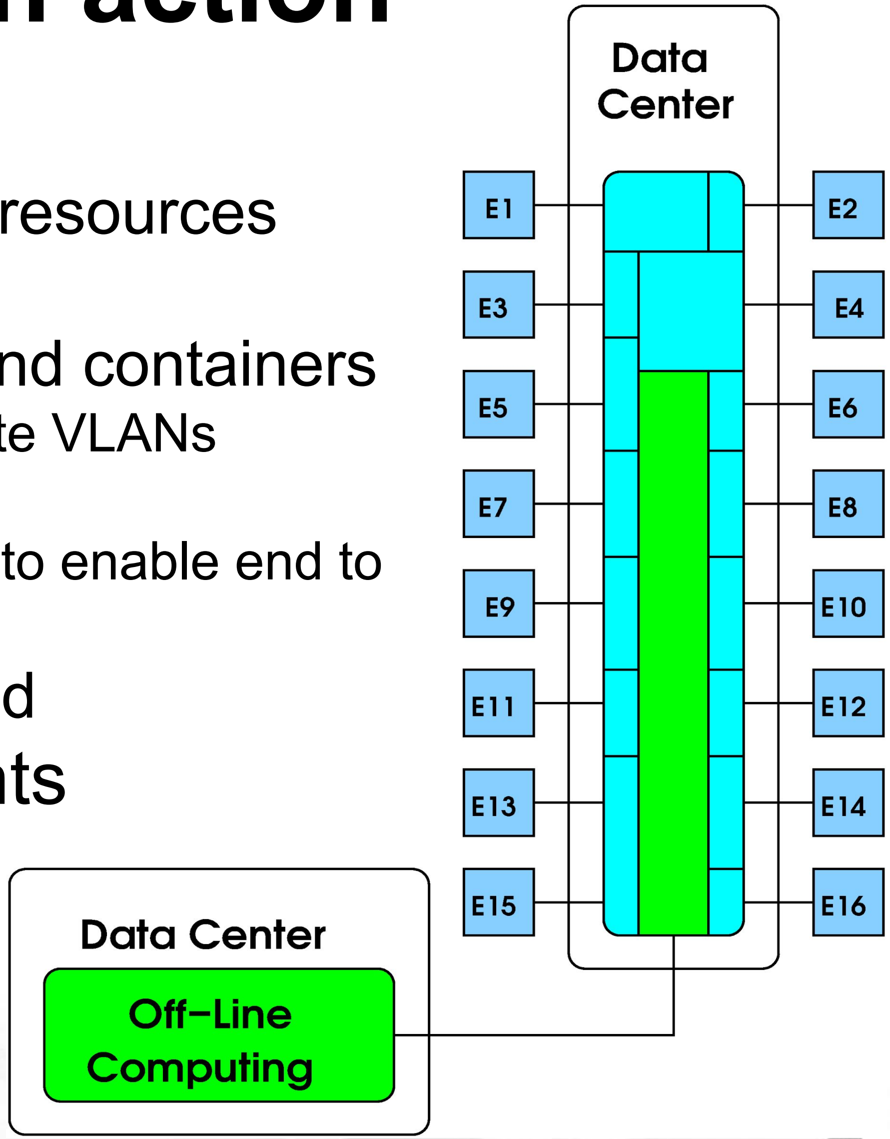
- Rack = Autonomous system (AS)
  - Effectively all nodes in rack visible as a block
  - Viable for bare metal or overlay network deployments
- Group of nodes in rack = AS
  - Nodes in ASn visible as a block
  - Requires Virtual Routing and Forwarding (VRF) capable TOR, one VRF per ASn
- Multi-tenant node = one AS per tenant
  - Groups of VMs or containers, one group per tenant
  - Requires VRF capable switch
- Multi-tenant node = VXLAN overlay network
  - May require VXLAN support on switch.

Network Resources

- External Network
  - Capabilities of external network control software not scoped out.
    - Overlay network (VXLAN) can probably be managed with cloud platform software
    - Unclear if cloud platform software can manage non-overlay network solutions.
  - Can static network configuration satisfy requirements ?
    - Necessary functionality may be possible by changing internal network configurations with no change to external network configuration.
      - e.g., static, pre-configured AS to AS connectivity; static, pre-configured VTEP endpoints.
  - If dynamic reconfiguration of network is required, does cloud platform software exist to manage the components ?
  - Further investigation is necessary

Virtual On-Line Computing in action

- Steps to augment on line resources
  - Experiment E1 (Expt E1) requests compute resources
  - Containers instantiated on chosen hosts
  - Connectivity established between Expt E1 and containers
    - Container network interfaces placed in appropriate VLANs
    - VRFs created for the VLANs
    - Route advertisements altered in the core routers to enable end to end connectivity
  - Reverse process when resources not needed
- Process is replicated for other experiments
- Unallocated resources used to augment off line computing resources



Virtual On-Line Computing in action

- Data from E1 can be streamed directly to the containers
  - No intervening firewall, DTN, or proxy
  - No need to write data to disk then send data from disk to container
- No VXLAN overhead if non-overlay routed network used.
- Containers/VMs only accessible from Expt E1
  - However, additional connectivity can be enabled if desired
  - Expt E1 protected from systems/users at the data center
- Data center protected from Expt E1
- If the process is completely automated and resource are available, containers can be instantiated quickly

Next Step

- Determine if the virtual online computing concept meets the actual needs of the community
- If yes, determine the path forward for the implementation of the system
  - Investigate the various software solutions
  - Determine the best fit for the problem at hand
  - Identify an missing components and develop solutions
- Requires close cooperation between compute, network, and cybersecurity teams