SENSE
(SDN for End-to-End Networking @ Exascale)
Project

Inder Monga
Lead PI

What Problem(s) are We Solving

- End-to-end network service automation
  - Manual provisioning
  - No service consistency across domains
  - No service visibility across domains
- Application-Network interaction missing
  - Ability for science workflows to drive service provisioning
  - Programming APIs usually not intuitive and require detailed network knowledge, some not pre-known
  - Detailed network information needed, usually not easily available

What Problem(s) are We Solving

- Multi-domain service visibility and troubleshooting
  - Data APIs across domains for applications, users, network administrators
  - Performance, service statistics, topology, capability etc.
  - Exchange of ‘scoped’ and authorized information
- Alignment with security policies @ the end-site

Core idea

End-to-end, multi-domain provisioning automation and resource orchestration

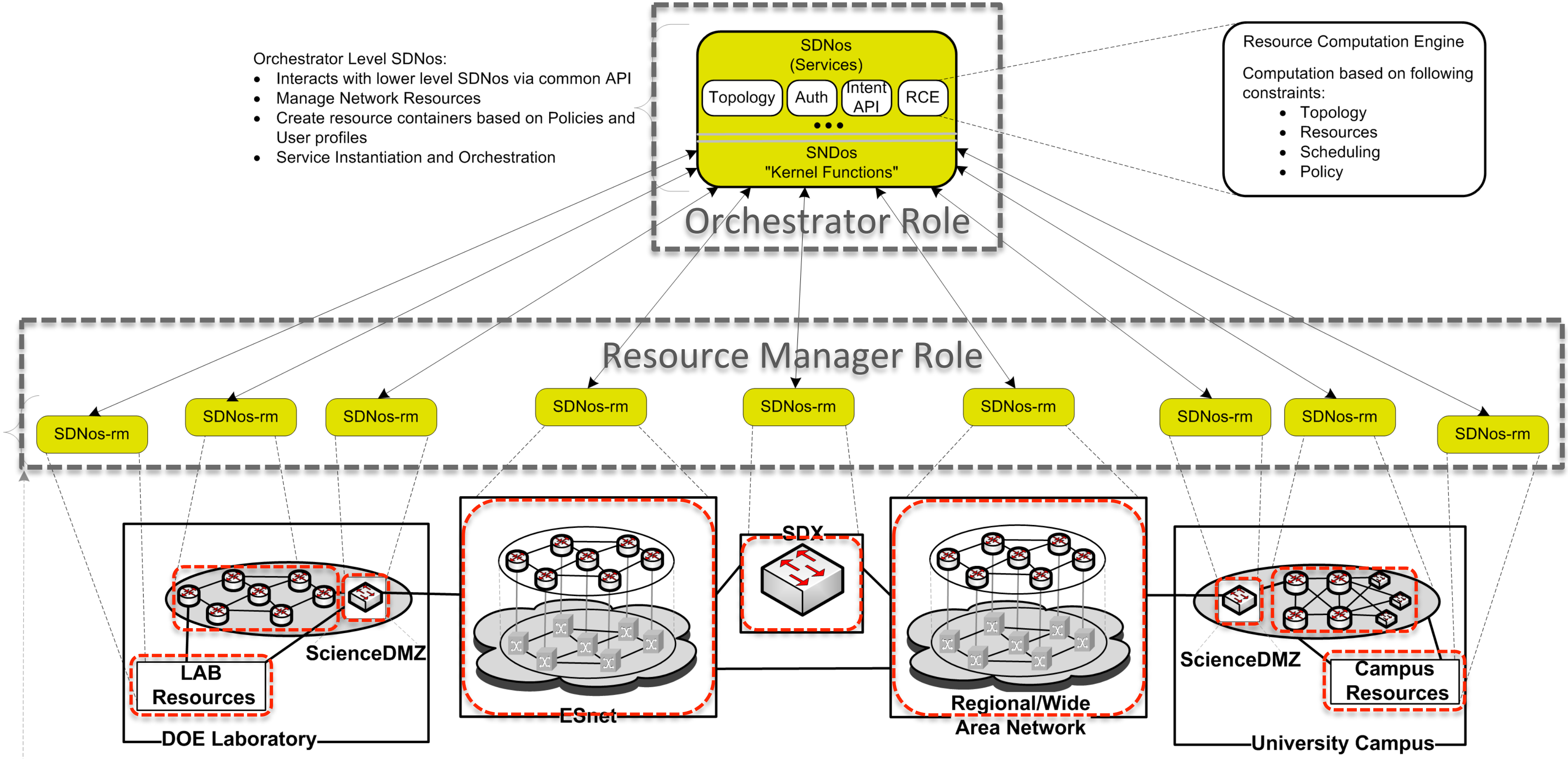
SENSE scoped definitions

- End-to-End (network point of view)
  - DTN NIC to DTN NIC, across Science DMZ, WAN(s), Open exchange points (ideally)
- Multi-domain
  - Multiple administrative domains, independent policies and AUP
- Provisioning automation
  - Bring-up and management of services without interrupt-driven human involvement
- Resource orchestration
  - Allocation and reservation of resources including compute, storage and network

Current State of SDN - Gaps

- Inter-domain SDN undefined
  - Not focus of open-source or commercial efforts
  - No organized R&E efforts
- Resource descriptions vary wildly across projects, industry
  - Minimal ethernet topology discovery by controllers, represented in proprietary JSON formats
- Multi-domain admission control and Authz
  - No clear way to specify AA, policies, and enforce them across domains
- A ‘usable’ multi-domain testbed

SENSE Architecture and Approach



Resource or Facility Specific SDNos
- Responsible for local resource of facility
- Implemenation system and technolgy a local decision
- Southbound APIs vary depending on resources/facility type
- Commmon Northbound API to be defined
- Resource descriptions based on extensions to NML

Defines Service Perimeter/Boundary

SDNos: SDN Operating System
SDNos-rm: SDN Operating System - Resource Manager

Why do we need a Network OS?

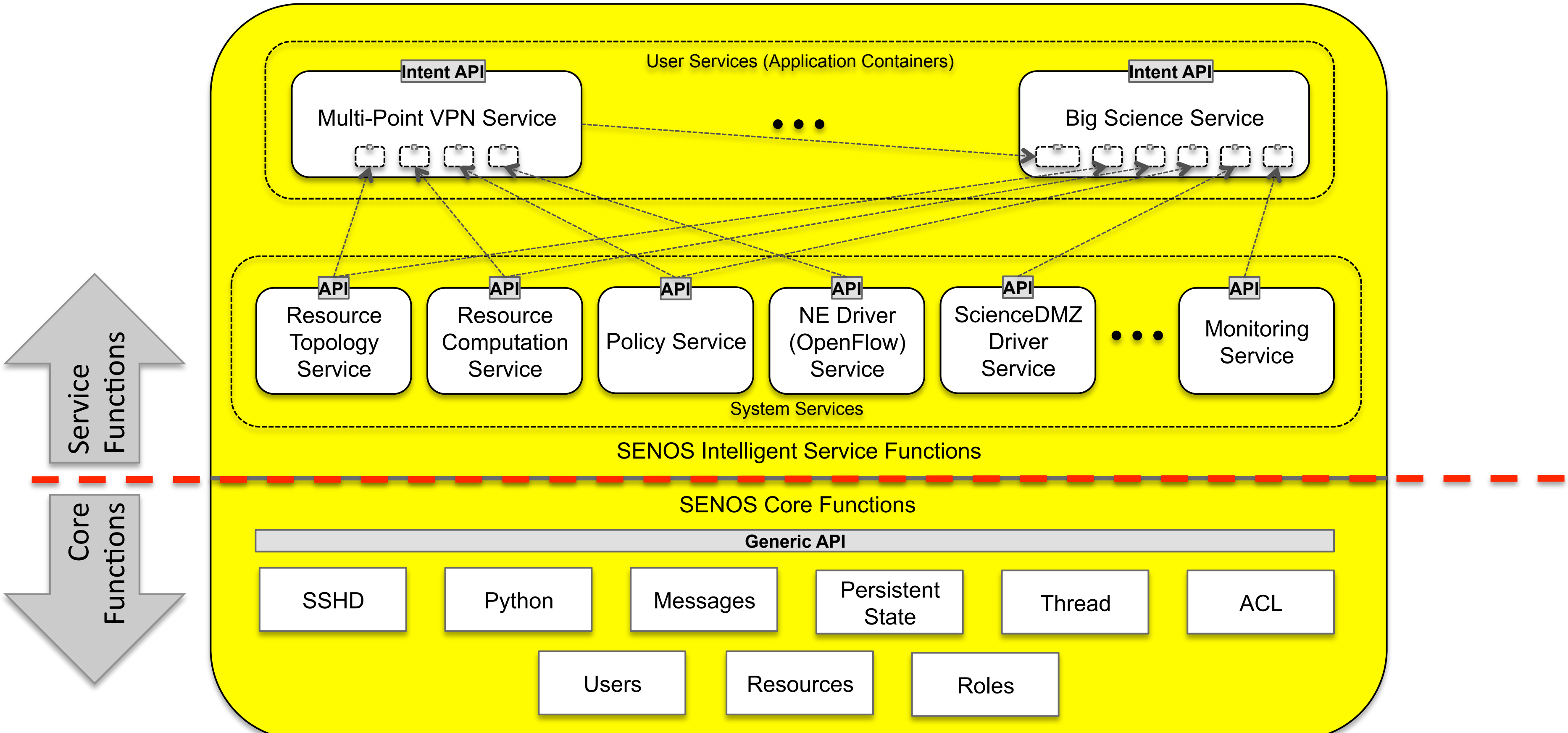
- Multiple application support
  - Missing from existing SDN controllers
  - Requires security, AA, policy infrastructure
- Ability for users to develop custom services
  - Using infrastructure services
  - Offer them to their own customers
- Resource sharing (not control) key
  - Network is shared by many tenants
  - Different service levels (Best Effort, Guaranteed, Low-latency etc.)

SENSE Requirements for an SDN OS

- Communication and coordination
  - between the control planes at different sites.
- Security mechanisms
  - to protect the integrity and availability of network, compute, and storage resources
- Abstractions of resource state and metadata
  - Consistent across multi-domain
- Express and Enforce local policies

SENSE OS (SENOS) Architecture

- Service functions provide the intelligence to interpret or render a user's intent, enforce policies, and coordinate workflows.
- Service functions are hierarchical in nature, with atomic services being a discrete set of services that can be composed to build a more complex, custom service.



- Separation of kernel (privileged) and multi-user (unprivileged) execution space
- Supports the concept of resources and allows the owners of resources to specify how those resources may be used, via access control lists.
- A set of generic object definitions for commonly-used objects in SDN programs, such as network nodes, ports, and links.
- An inter-process message-passing facility for communication between SENOS instances.

SENOS Intelligent Service Functions (1/2)

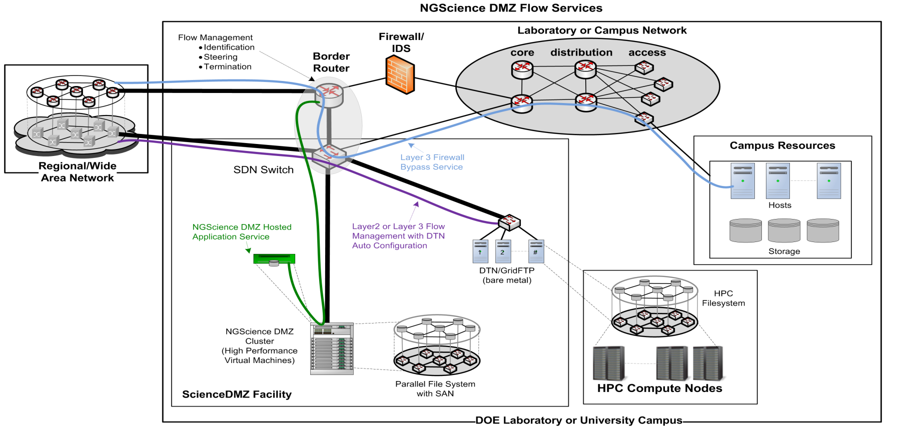
- Resource Information Service (RIS)
  - Harvest and normalizes resource topology
  - Enforces policy views of topology
- Resource Computation Service (RCS)
  - Multi-Constrain Multi Resource computation (from RAINS)
  - Add support for SENSE requirements
    - Next-Generation Science DMZ and site resources
    - Flow management, flow termination, and site services integration functions
    - Interaction with SENOS Policy Service

SENOS Intelligent Service Functions (2/2)

- Intent and Rendering
  - Intent APIs expose very high-level service abstractions that focuses on the ‘what’ the application wants to accomplish and not on the ‘how’
  - Renderer implements the business logic for the service
    - Authorization
    - Policy enforcement
    - Computation to determine a next set of tasks
    - Creation of subordinate (lower-layer) intents, if any
    - Directly performing a set of actions
    - Returning request status
  - Multi-Point VPN Service will be the first network service prototyped for the SENSE project

End-Site Orchestration

- Science DMZ Flow Management
  - Route to right flows to the right DTNs, vlan or more granular flow identification using OF
  - Support multi-science Science DMZ, with resource allocation and traffic steering
  - Enable addition of NFV services like Caching, and flow service chaining



End-Site Orchestration (contd.)

- DTN Autoconfig
  - Systems configuration to ensure data transfer application can connect @ Layer 3 and/or Layer 2
    - Includes, VLAN configuration on the NIC
    - Private or public IP address configuration of L2 or L3 VPNs
    - Other configuration like TCP window size, might be a stretch
  - Creating VMs or containers with right data-movement software for multi-science DTNs
  - OVS configuration and QoS configuration
  - Flow steering and ACLs to connect to the internal file system over different NICs

WAN / Regional / Exchange Network Orchestration

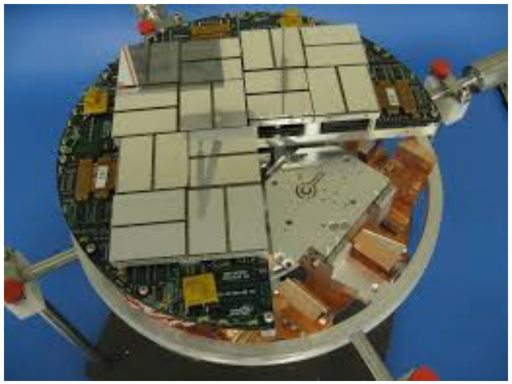
- Leverage past experience with dynamic point-to-point circuit services to develop multi-point offering
- Network Element (NE) Control
  - SENOS NE Driver for southbound communication (i.e. OpenFlow, NetConf/YANG, P4, CLI, etc)

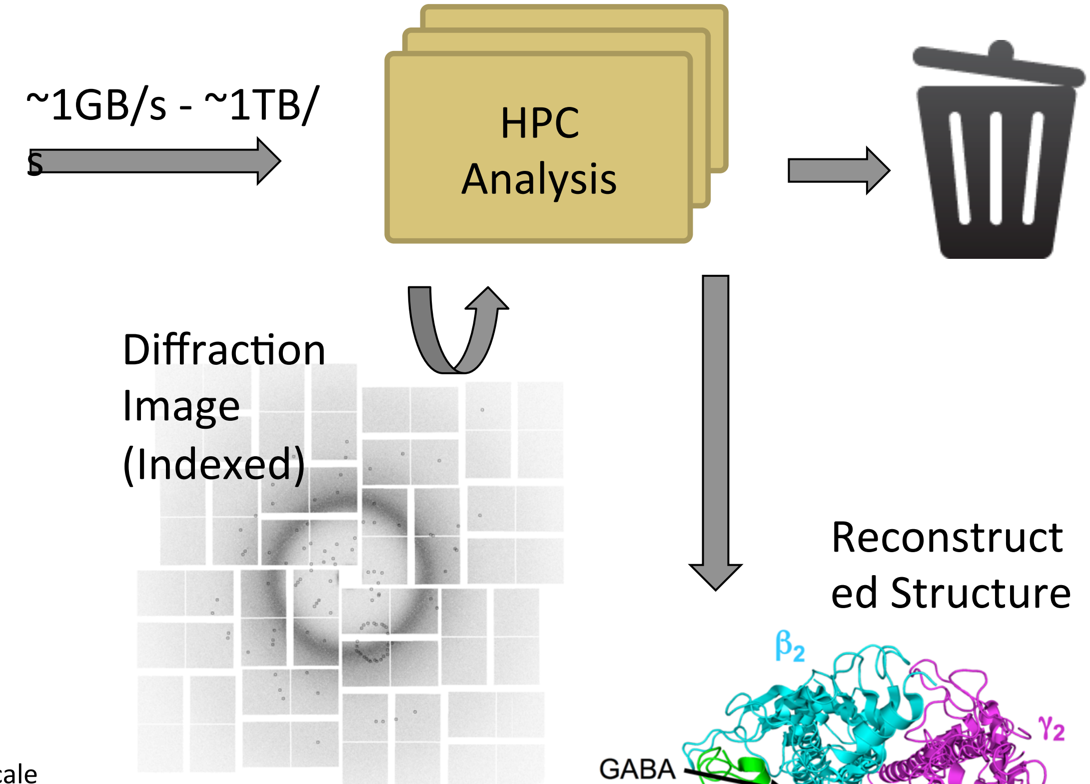
SENSE End-to-End Orchestration

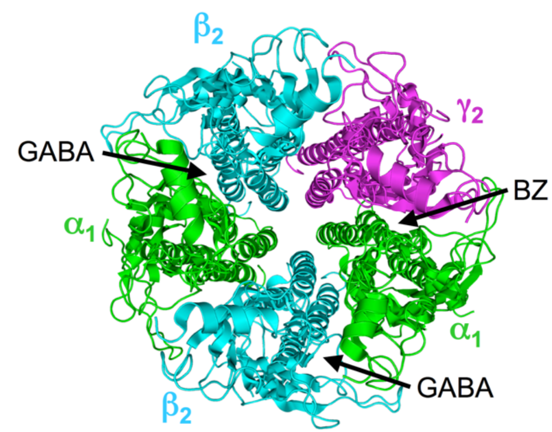
- Three Science Use-cases
  - LHC CMS use-case
  - NERSC Burst Buffer use-case
  - Superfacility use-case

LCLS Computing Use Case:
Quasi Real Time Nanocrystallography Pipeline

Providing atomic-scale vision to researchers at the beamline in < 10s





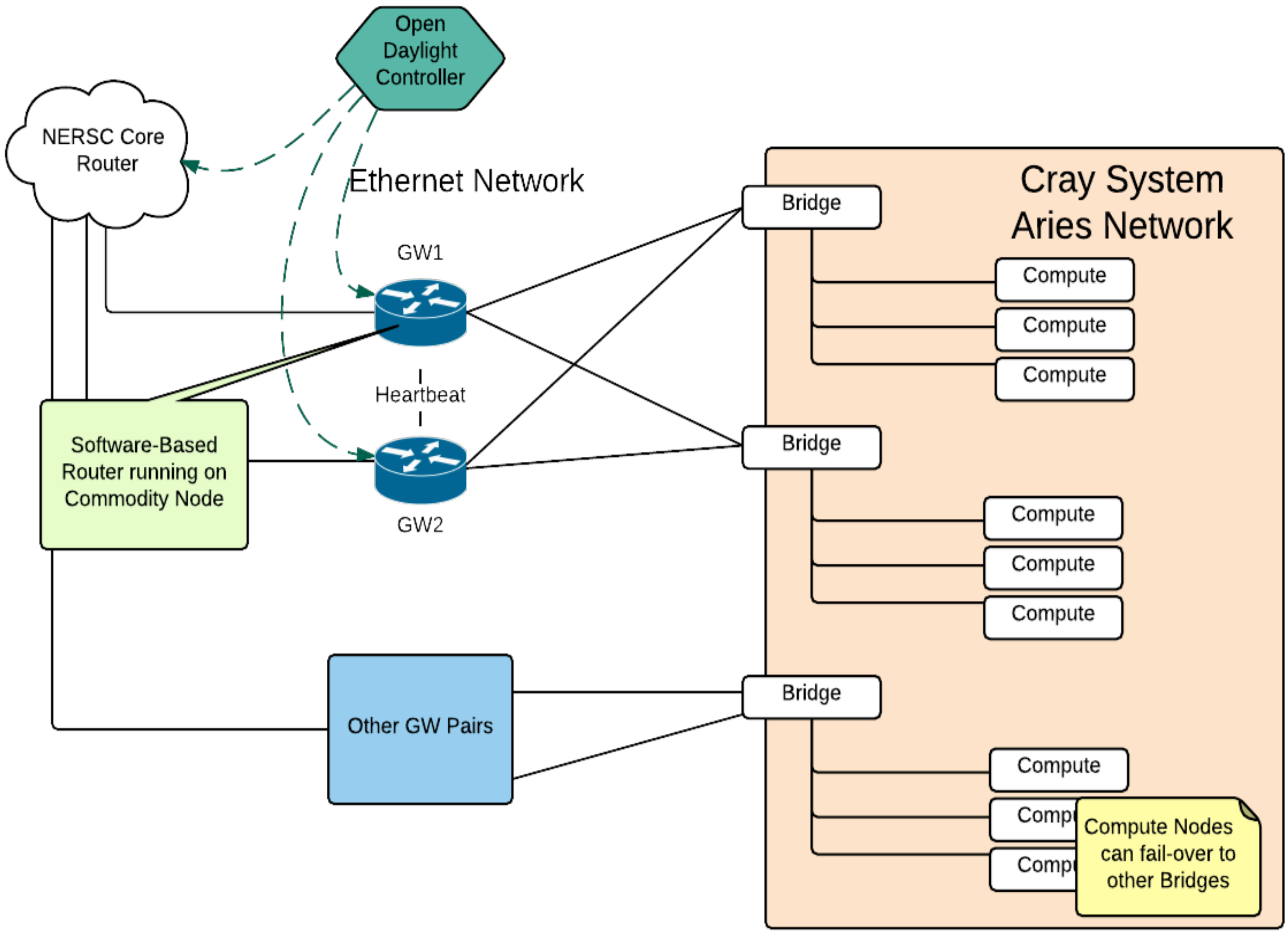


Streaming data from the detector to scalable HPC

- Indexing, classification and reconstruction
- Quasi real-time response (<10s)
- Currently (LCLS-I) requiring 50 TFLOPS
- Cori Supercomputer: 500K cores, 28 PB disk

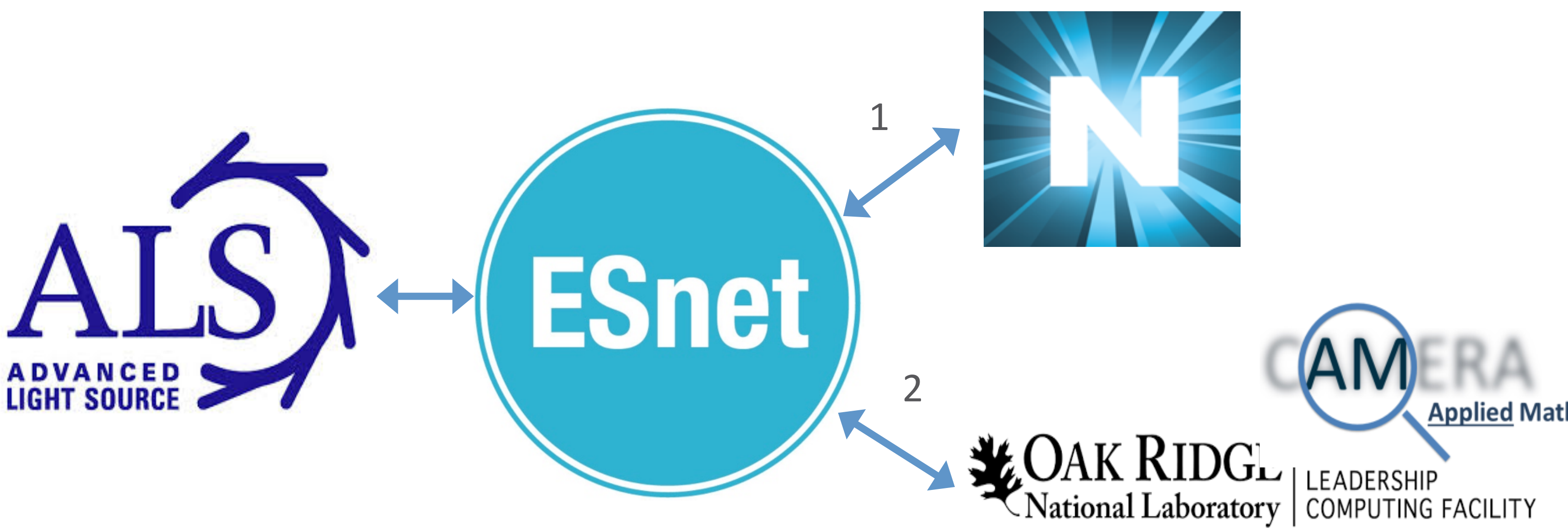
Pipeline critical for experiments studying atomic scale structural dynamics and fluctuations in matter:

- Complex materials (novel functional properties)
  - Heterogeneity and fluctuations at the nano-scale
  - Nano-particle dynamics
- Catalysis (efficient, selective, robust, earth-abundant)
  - Chemical, structural, and electronic changes; Nano-particles; Interfacial chemistry
- Biological function
  - Protein crystallography – structure and dynamics from reconstructions
  - Macro-molecules – conformational dynamics, heterogeneity, and interacting bio complexes


Open Daylight Controller
NERSC Core Router
Ethernet Network
GW1
Heartbeat
GW2
Software-Based Router running on Commodity Node
Other GW Pairs
Cray System Aries Network
Bridge
Compute
Compute
Compute
Bridge
Compute
Compute
Compute
Bridge
Compute
Compute Nodes can fail-over to other Bridges

Superfacility Prototype and Use Case : Process of science transformed

Real-time analysis of 'slot-die' technique for printing organic photovoltaics, using ALS + NERSC (SPOT Suite for reduction, remeshing, analysis) + OLCF (HipGISAXS running on Titan w/ 8000 GPUs).

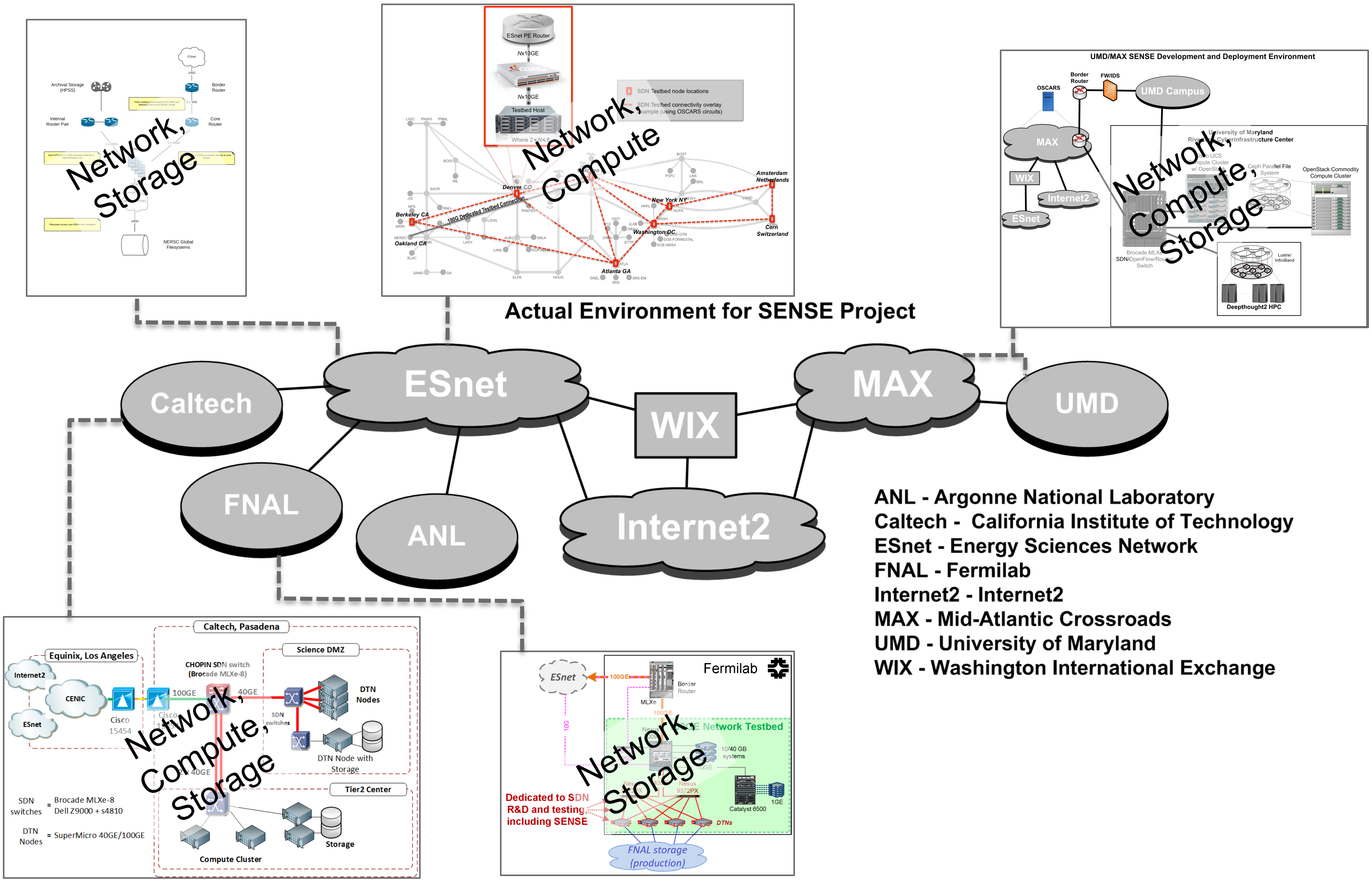


http://www.es.net/news-and-publications/esnet-news/2015/esnet-paves-way-for-hpc-superfacility-real-time-beamline-experiments/ Results presented at March 2015 meeting of American Physical Society by Alex Hexemer. Additional DOE contributions: **GLOBUS** (ANL), **CAMERA** (Berkeley Lab)

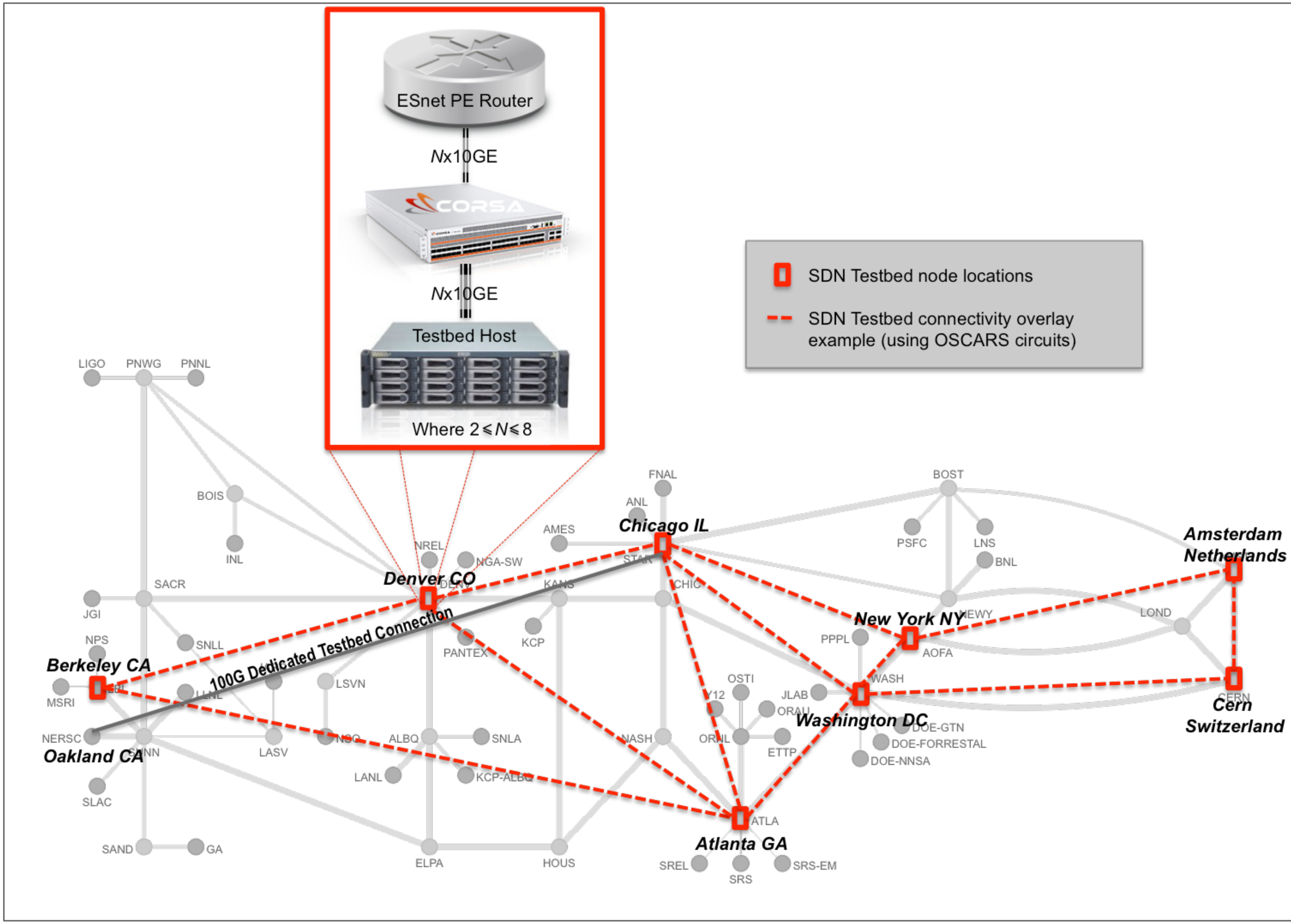
Instrumentation / Monitoring / Measurements

- For the user:
  - Service specific data to the user
  - Follows the thought behind the intent API, abstracted service metrics
- For the administrator:
  - More detailed information across all user-services
  - Multi-domain information exchange to help debug/troubleshoot/ monitor service levels
  - Continuous monitoring
- Data Analysis
  - Data mined will be used for system modeling and machine learning to perform capacity predictions and develop strategies for negotiating resources
  - This task will collaboratively lead by the SDN NGenIA project

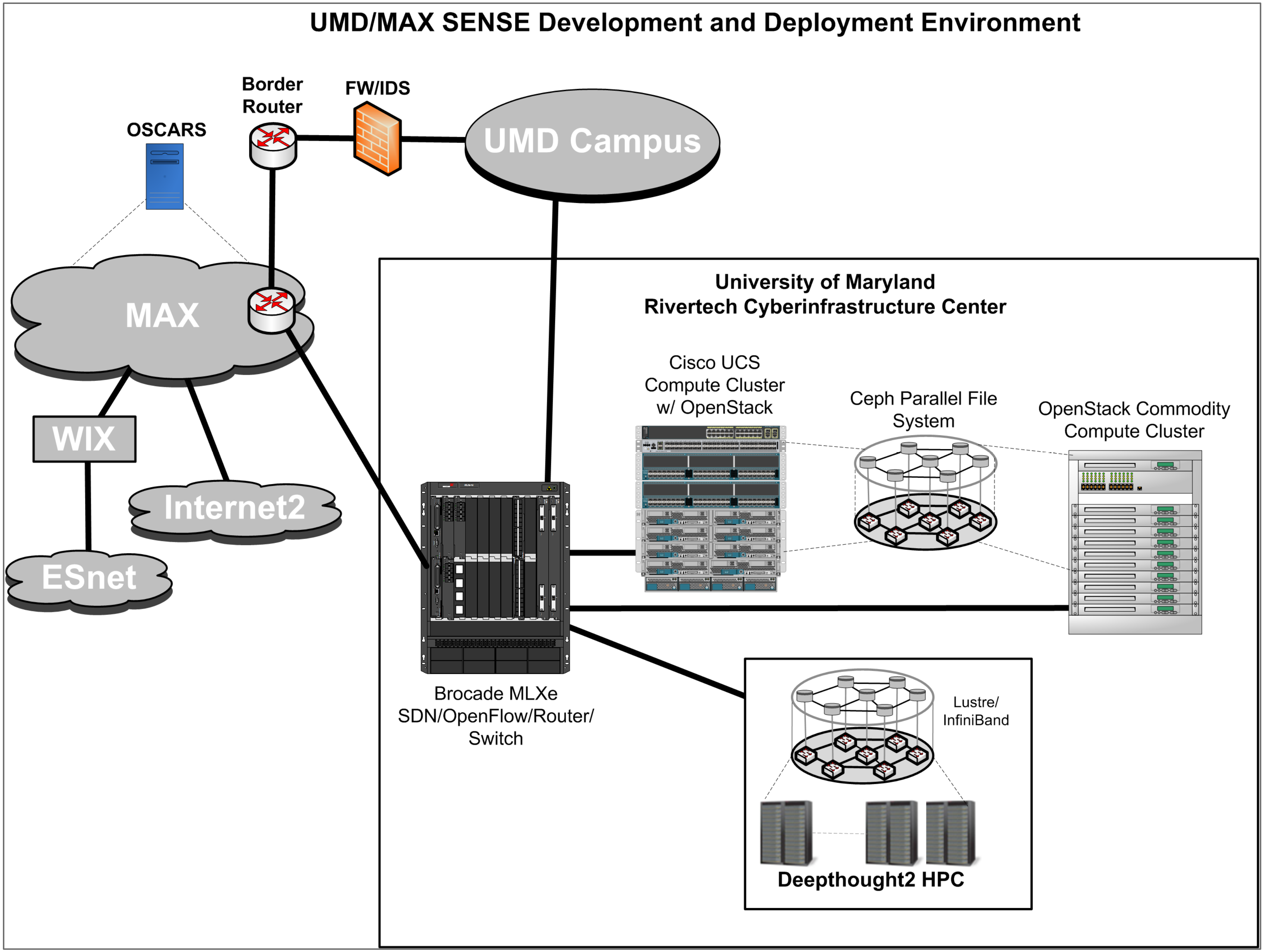
End-to-End SENSE Testbed

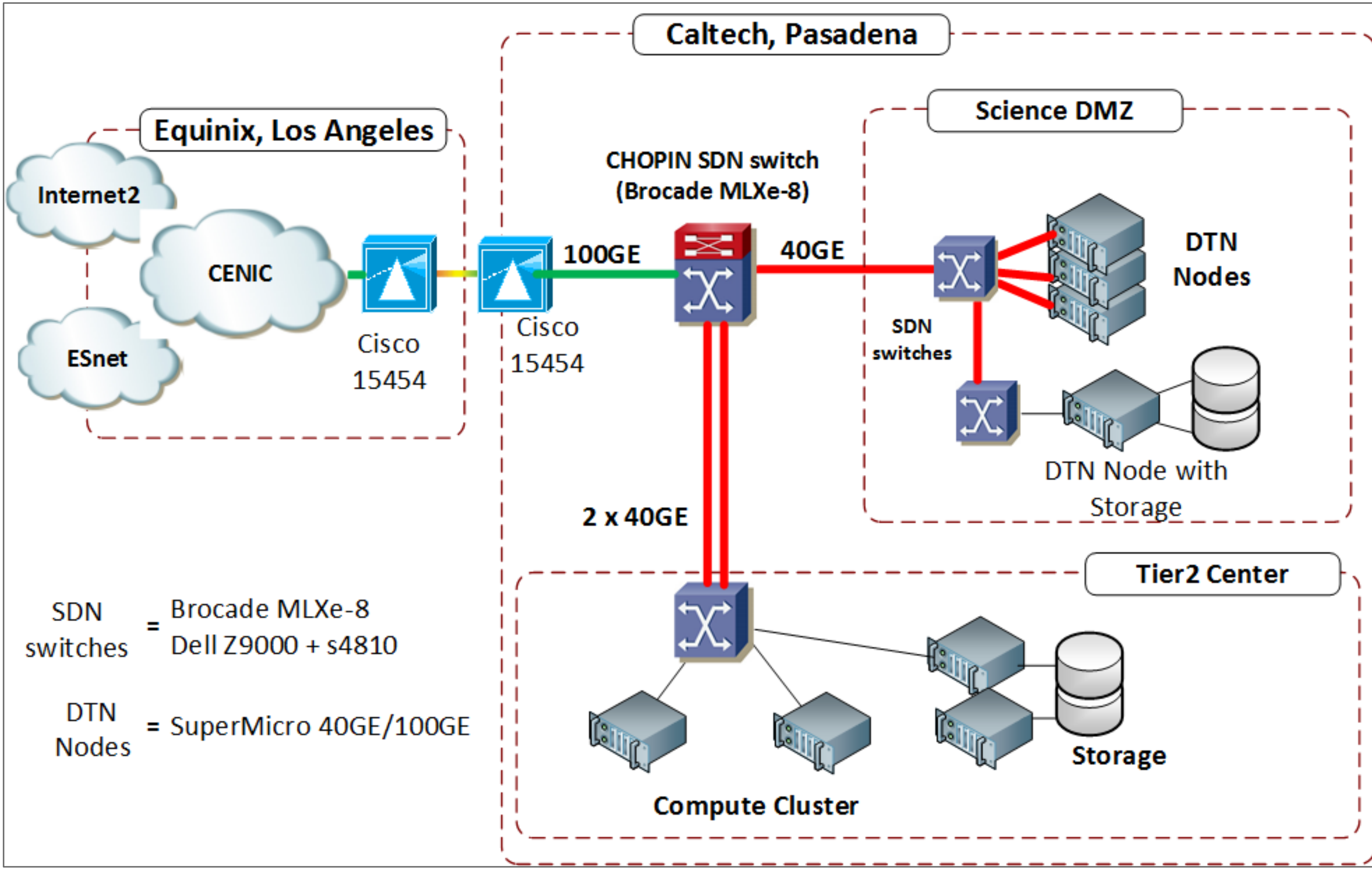


ANL - Argonne National Laboratory
Caltech - California Institute of Technology
ESnet - Energy Sciences Network
FNAL - Fermilab
Internet2 - Internet2
MAX - Mid-Atlantic Crossroads
UMD - University of Maryland
WIX - Washington International Exchange

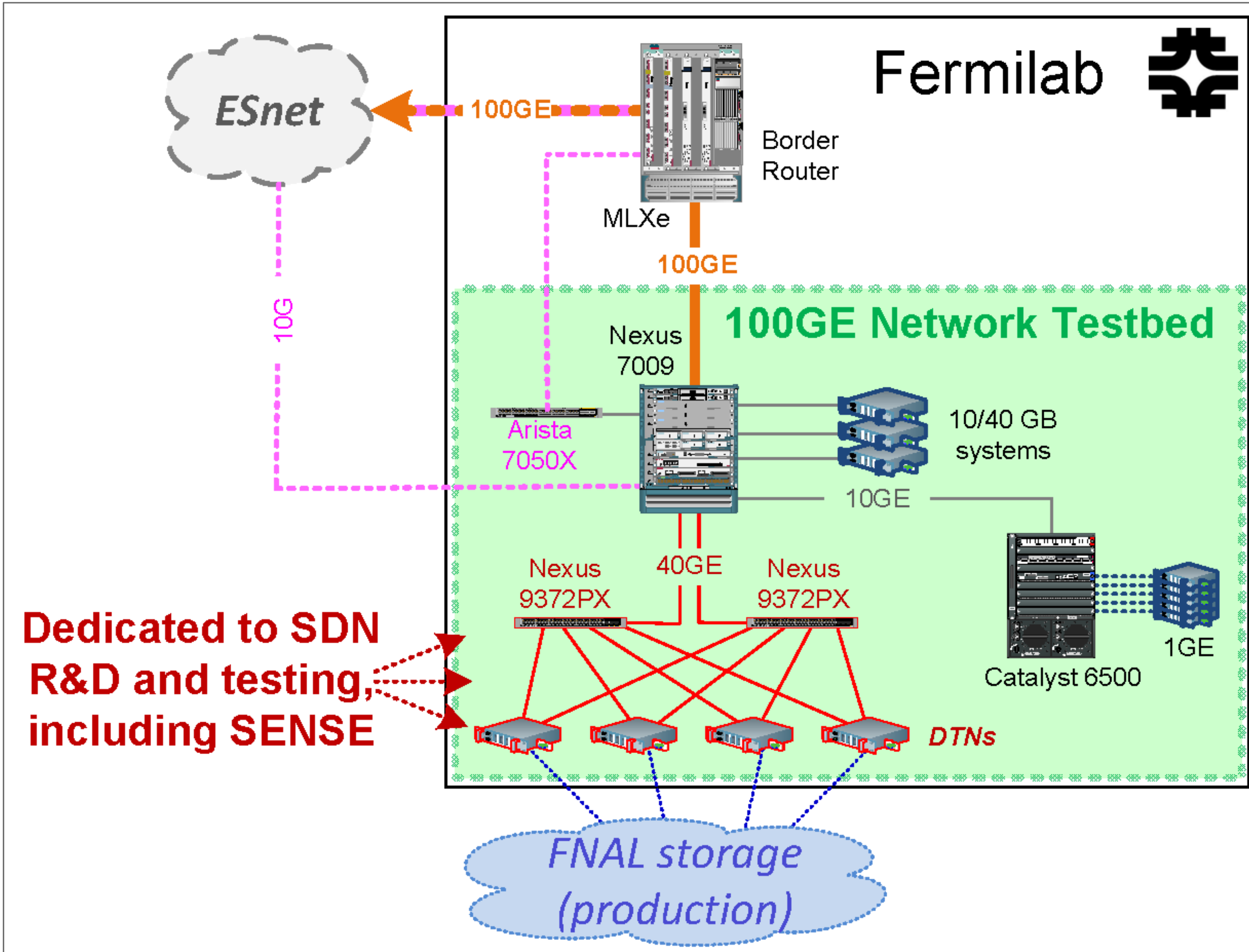


ESnet PE Router
Nx10GE
CORSA
Nx10GE
Testbed Host
Where 2 ≤ N ≤ 8
SDN Testbed node locations
SDN Testbed connectivity overlay example (using OSCARS circuits)
100G Dedicated Testbed Connection
Berkeley CA
Oakland CA
Denver CO
Chicago IL
New York NY
Washington DC
Atlanta GA
Amsterdam Netherlands
Cern Switzerland
LIGO
PNWG
PNNL
BOIS
INL
SACR
JGI
NPS
SNLL
MSRI
NERSC
SLAC
SAND
GA
LSVN
LASV
ALBQ
SNLA
LANL
KCP-ALBQ
ELPA
HOUS
NREL
NGA-SW
PANTEX
KCP
AMES
KANS
FNAL
ANL
STAR
CHIC
NASH
OSTI
Y12
ORNL
ORAU
JLAB
ETTP
ATLA
SREL
SRS
SRS-EM
BOST
PSFC
LNS
BNL
NEWY
AOFA
PPPL
WASH
DOE-GTN
DOE-FORRESTAL
DOE-NNSA
LOND
CERN

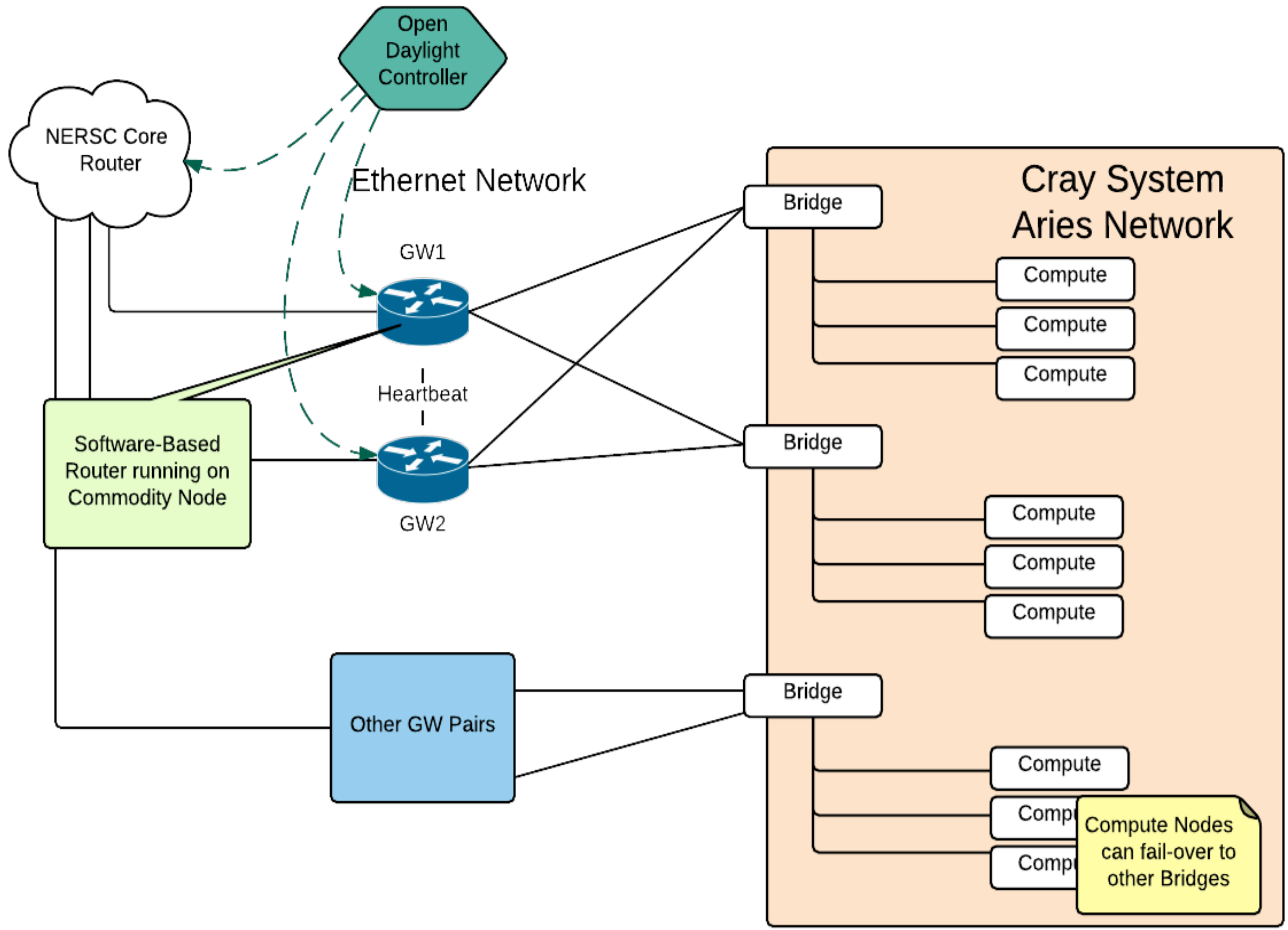

UMD/MAX SENSE Development and Deployment Environment
Border Router
FW/IDS
OSCARS
UMD Campus
MAX
WIX
Internet2
ESnet
University of Maryland
Rivertech Cyberinfrastructure Center
Cisco UCS Compute Cluster w/ OpenStack
Ceph Parallel File System
OpenStack Commodity Compute Cluster
Brocade MLXe SDN/OpenFlow/Router/ Switch
Lustre/ InfiniBand
Deepthought2 HPC



Caltech, Pasadena
Equinix, Los Angeles
Science DMZ
Internet2
CENIC
ESnet
Cisco 15454
Cisco 15454
100GE
CHOPIN SDN switch (Brocade MLXe-8)
40GE
DTN Nodes
SDN switches
DTN Node with Storage
2 x 40GE
Tier2 Center
Storage
Compute Cluster
SDN switches = Brocade MLXe-8
Dell Z9000 + s4810
DTN Nodes = SuperMicro 40GE/100GE



ESnet
100GE
Border
Router
MLXe
Fermilab
100GE
10G
100GE Network Testbed
Nexus
7009
Arista
7050X
10/40 GB
systems
10GE
40GE
Nexus
9372PX
Nexus
9372PX
1GE
Catalyst 6500
Dedicated to SDN
R&D and testing,
including SENSE
DTNs
FNAL storage
(production)


Open Daylight Controller
NERSC Core Router
Ethernet Network
Cray System
Aries Network
Bridge
GW1
Compute
Compute
Compute
Heartbeat
Software-Based Router running on Commodity Node
Bridge
GW2
Compute
Compute
Compute
Other GW Pairs
Bridge
Compute
Compute Nodes can fail-over to other Bridges

Flying Start: Leveraging prior work

- Modeling
  - RAINS project
- Testbed
  - Existing compute and DTN equipment @ sites
  - ESnet SDN testbed across US/Europe
  - GENI resources
- SDN Software
  - ENOS and ODL concepts and some components from ESnet/Harvey

Deliverables: Year 1 (Feb* – Feb)

- Science Use-Cases requirements
  - Influence the design and site implementation
- SENOS Architecture and Design, overall
  - With initial implementation of SDN software components
- Intent Interfaces
  - Initial implementation for Multi-point VPN and DTN Autoconfiguration
  - Design for service metrics feedback to application
- NG Science DMZ
  - Design and architecture
  - Initial implementation of SENOS for Local Science DMZ network

* Key SENSE members haven't received funding yet

Deliverables: Year 2

- Implementation of SDN Science DMZ @ end-sites
- Enhance the SDN testbed with features needed for SENSE
- Final implementations of
  - Intent Rendering
  - SENOS
  - SENSE Orchestrator
- Integration into Science Workflows

Deliverables: Year 3

- Demonstration of Science Workflows/Use-cases over the testbed
- Improved Intent Interfaces
- Operational information sharing between sites
- Final implementation of SENOS components
- Tech Transfer and transition

Management Plan discussion

- Monthly PI meetings: Chin/Inder
  - Progress, issues
- Quarterly Progress reports in bullets
- Website for information sharing, both internal and external to project
- Software on Github

Roles and Responsibilities shared across the team

- Design/Architecture: Chin/Inder
  - Define/Write requirements for the software features, and WAN, LAN, Science implementation and science use-cases
  - Multi-domain orchestration: **Tom Lehman**
    - Resource models, design, AA, etc.
  - Policies and End-Site Orchestration**: Phil DeMar**
  - Site policies, ScienceDMZ automation, end-host automation requirements
- Software: John Macauley
  - Processes, release mechanism, documentation, web page etc.
- Deployment: Linda Winkler
  - Deployment of SDN testbed, architecture, timeline
- Science Use-cases & demos: Harvey/Azher
  - Coordination of science use-cases, demos etc.
- Outreach: TBD
  - News, Bullets, Website, Wiki, Logo
- Overall Project Management: Inder/Chin
  - Work with Thomas, deliverables tracking, reports etc.

Unified Topology Views Needed for Computation and Service Reasoning

- Multi-Resource, Full Stack
- End-to-End R&E path

RAINS – Resource Aware Intelligent Network Services