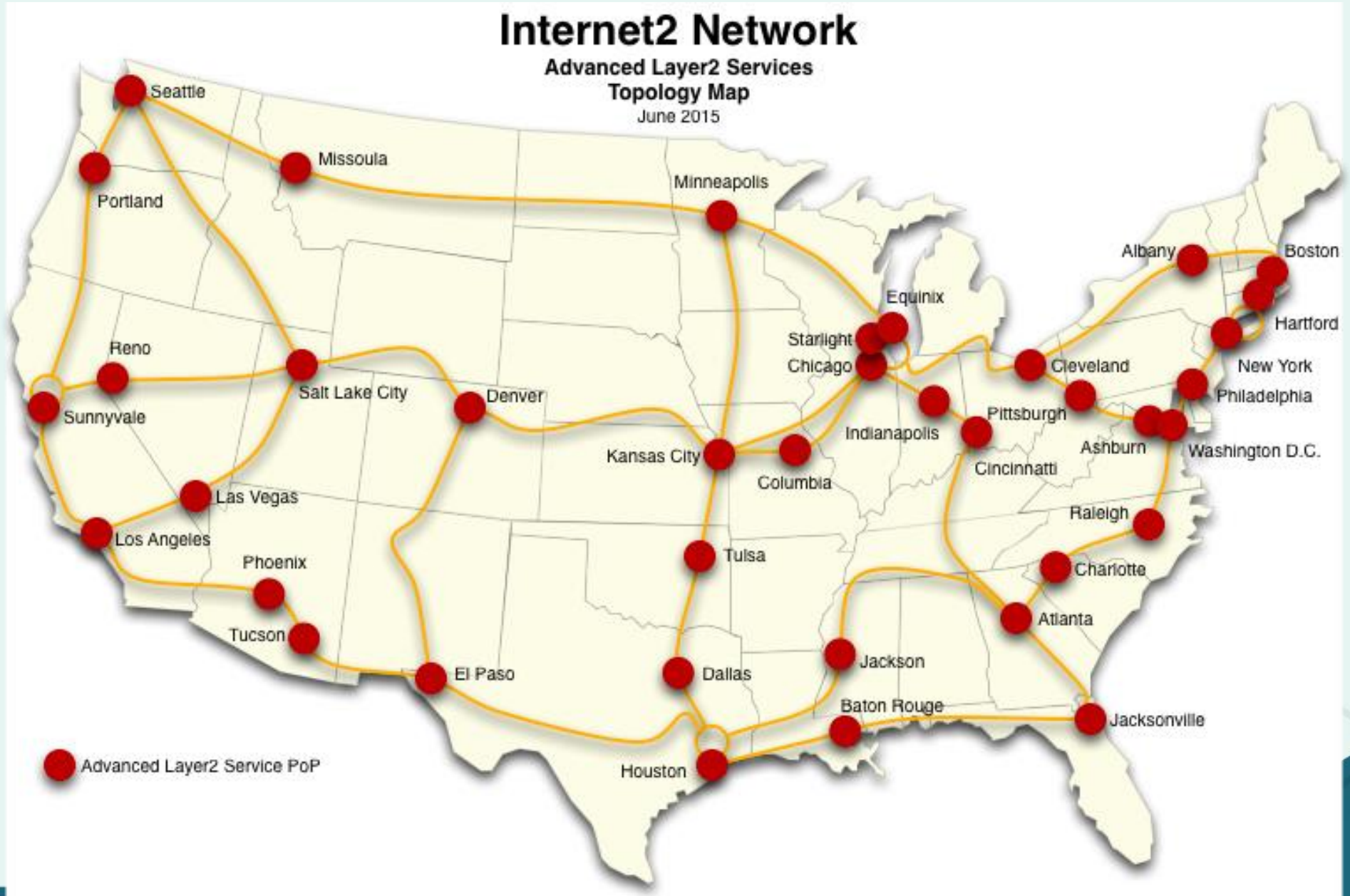
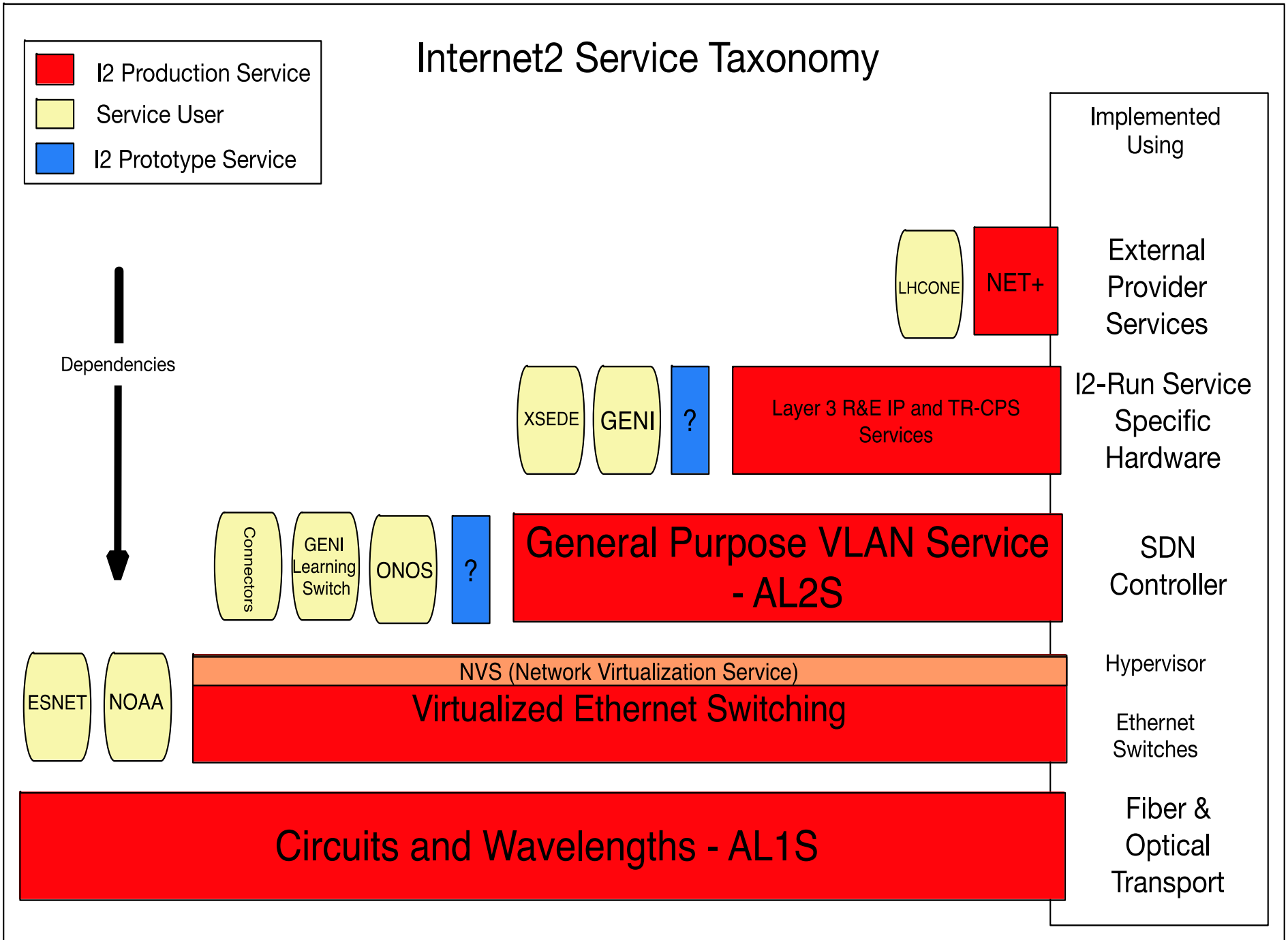
JOHN HICKS

Network Research Engineer

SEPTEMBER, 2015

Innovation and Experimentation through SDN and Network Virtualization


Internet2 Network
Advanced Layer2 Services
Topology Map
June 2015
Seattle
Portland
Missoula
Minneapolis
Albany
Boston
Equinix
Hartford
Starlight
Chicago
Reno
Salt Lake City
Denver
Cleveland
New York
Philadelphia
Sunnyvale
Pittsburgh
Indianapolis
Ashburn
Washington D.C.
Kansas City
Columbia
Cincinnati
Las Vegas
Raleigh
Los Angeles
Tulsa
Phoenix
Charlotte
Atlanta
Tucson
Jackson
El Paso
Dallas
Baton Rouge
Jacksonville
Houston
Advanced Layer2 Service PoP


I2 Production Service
Service User
I2 Prototype Service
Internet2 Service Taxonomy
Implemented Using
Dependencies
LHCONE
NET+
External Provider Services
XSEDE
GENI
?
Layer 3 R&E IP and TR-CPS Services
I2-Run Service Specific Hardware
Connectors
GENI Learning Switch
ONOS
?
General Purpose VLAN Service - AL2S
SDN Controller
ESNET
NOAA
NVS (Network Virtualization Service)
Virtualized Ethernet Switching
Hypervisor
Ethernet Switches
Circuits and Wavelengths - AL1S
Fiber & Optical Transport

Internet2 Goals

- Support production networking
  - Across Internet2
  - Integrated with partner networks around the globe
- Be a leader in advanced networking
  - Show what can be done, don't copy what commercial world is doing
- Innovation Platform:
  - Abundant Bandwidth (100G +)
  - Deeply Programmable (SDN)
  - Friction Free Science (Science DMZs)
- Use Cases …
  - Domain scientists collaborating in data intensive science
  - Extending local control over far-flung campuses (e.g. US, China, Middle East)
  - Massively Online Courses
  - Etc.

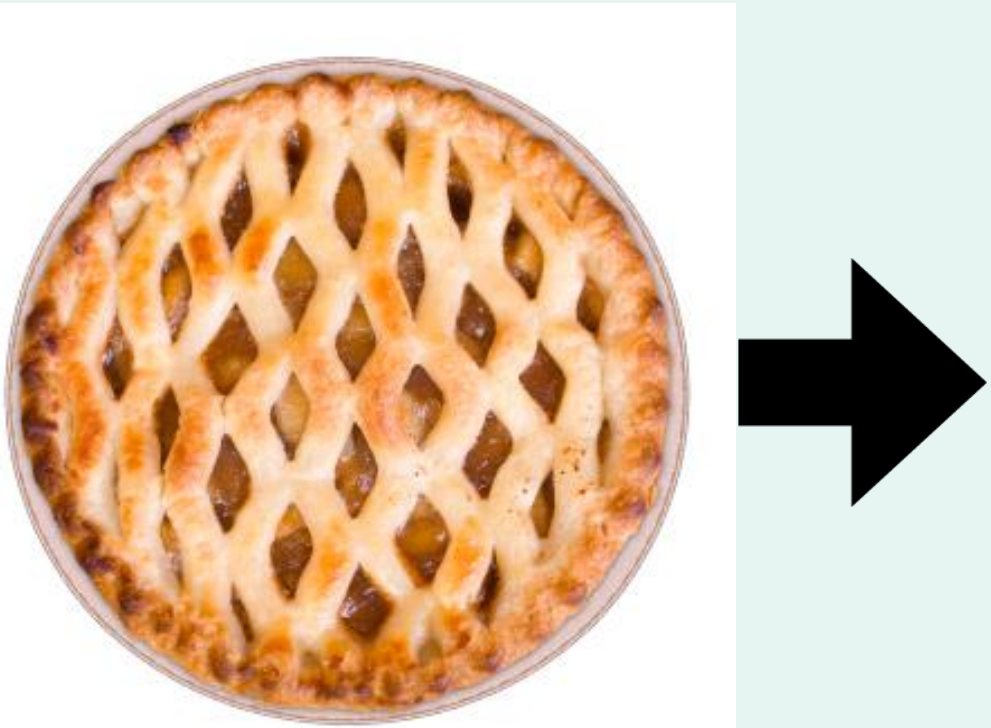
Internet2 Philosophy

- Support open, standards-compliant implementation of SDN
  - Strongly resist vendor impulse towards vertical integration
  - Decoupling control plane from data plane enables competition between switch vendors and competition between controllers
- Deploy multiple vendors in a common network to force inter-operability
  - Creates a “lowest common denominator” effect
  - Reflects the reality of the R&E networking community

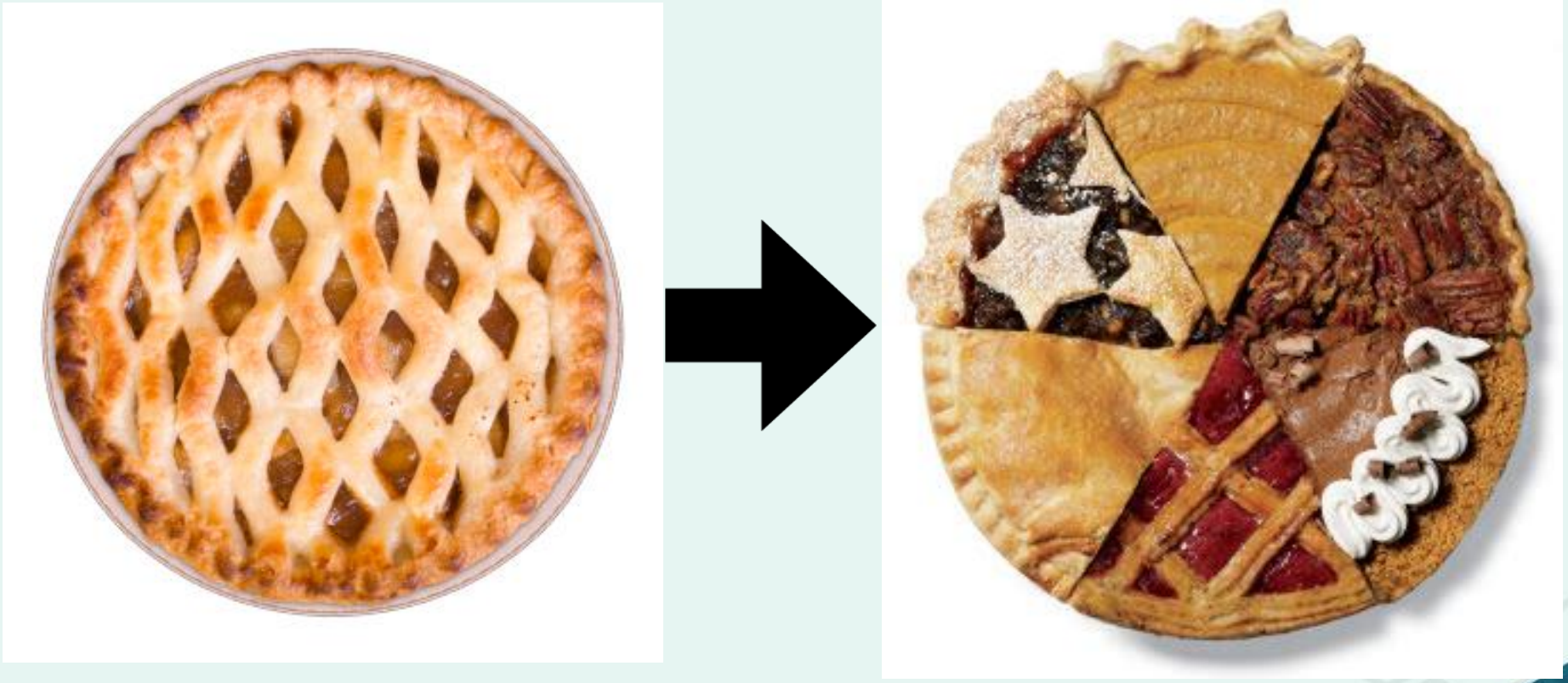
    5-7 networks along the E2E path
- Support both Production Networking and Innovation
  - Twin goals are definitely at odds but …
  - What is the point of R&E networks if we’re following?
  - There isn’t financial support to build operational-quality R&E networks just for network research
- Harness the strengths of R&E community to influence the market
  - Open, collaborative, innovative community
  - Collectively we have the power to change the conversation

Network Virtualization on Internet2

Network Virtualization on Internet2



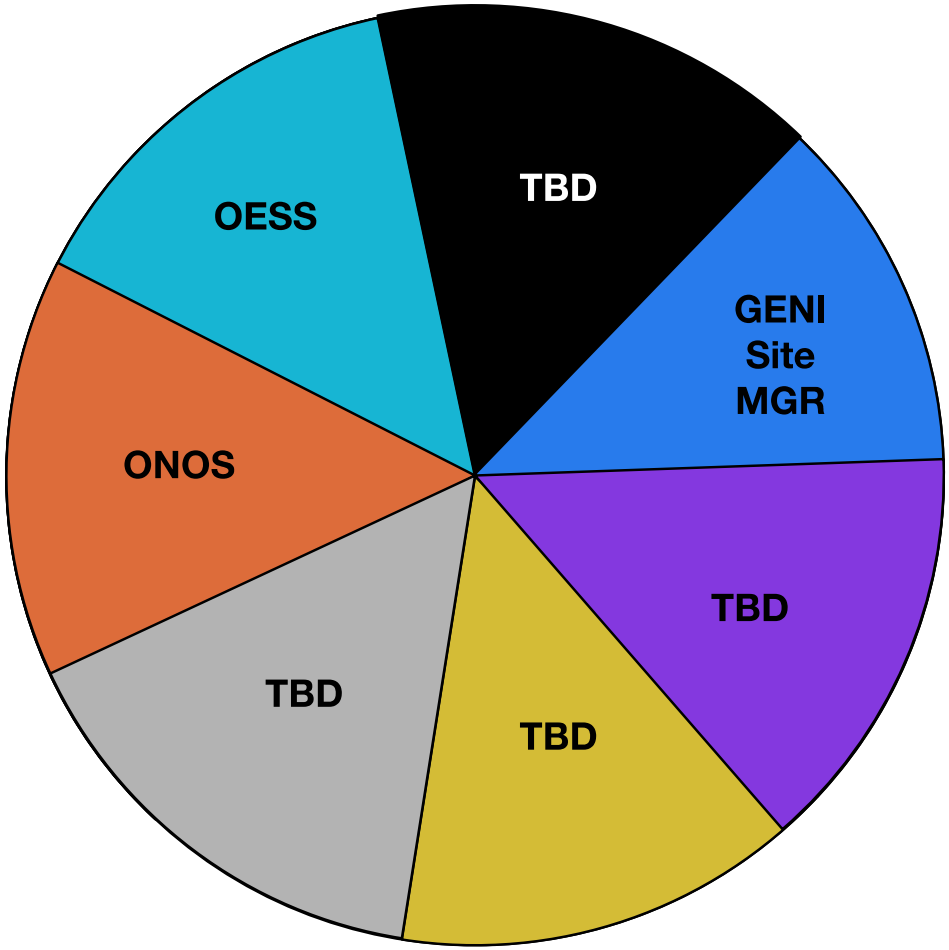
Network Virtualization on Internet2



Network Virtualization on Internet2

- Control a slice of the national network!
- Enable:
  - Rapid prototyping of advanced applications
  - Rapid prototyping of new network services
  - Rapid advancement of network research

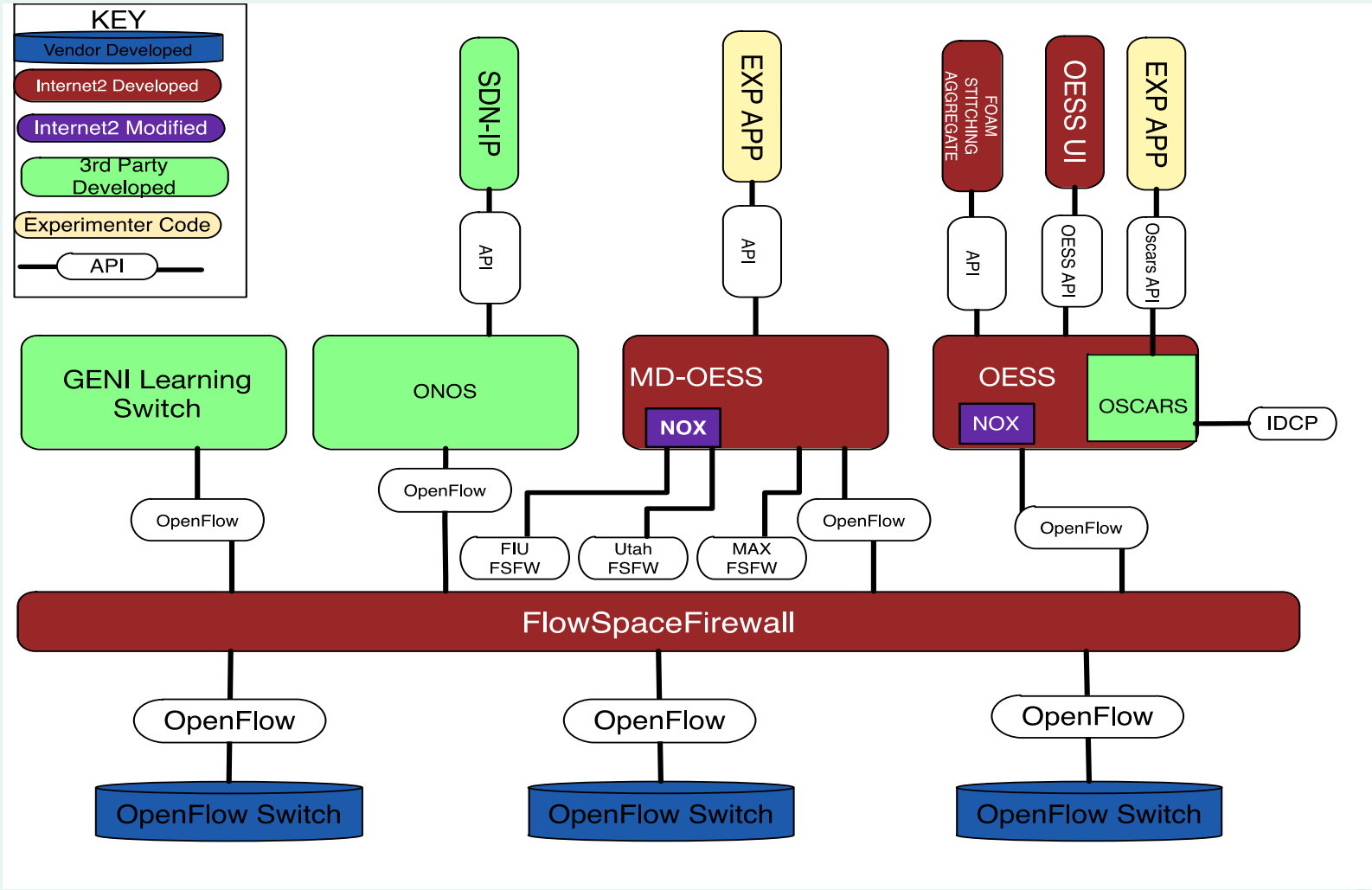
Network Virtualization on Internet2



Notable Milestones to Date

- April 2012: Internet2 announces intent to build 100G Layer 2 network on an SDN substrate in partnership with Indiana University
- October 2012: Internet2 AL2S launched on Brocade MLXe-16s in pure OpenFlow mode: First nationwide, open 100G network built on SDN Substrate
- March 2013: Internet2 AL2S becomes multi-vendor with introduction of Juniper MX-960s in pure OpenFlow mode
- May 2013: Juniper OpenFlow implementation becomes fully supported
- December 2013: Mulitpoint VLANs supported
- June 2014: Network Virtualization implemented through Flowspace Firewall hypervisor
- August 2014: Partnership with ON.LAB begins
- October 2014: GENI Sitemon v0.1 becomes first “alien” controller running on the Internet2 network
- October 2014: Multi-Domain SDX demonstrated
- April 2015: ONOS Controller / SDN-IP demonstrated with 3 universities
- June 2015: Three continent deployment of router-less Layer 3 network using ONOS and SDN-IP

Internet2 Network Software Stack



Internet2 September, 2015 Current Status

- OpenFlow 1.0 in production
  - OF 1.3 support in FSFW and OESS in design
  - Experimenting with Brocade 5.8b implementation of OF1.3
  - Working with Juniper on implementing requirements for OF 1.3
- Hypervisor (FlowSpace Firewall1.0.5) in production
  - Supports L2 and L3 matching
  - Vendor Updates (current versions Juniper 13.3, Brocade 5.6dc)
  - Vendor-specific limits do exist.
- Controller (OESS 1.1.6) in production
  - Supports Layer 2 Trace
- Accepting 3rd party controllers
  - Questionnaire
  - Mininet Emulation of AL2S
  - Test Lab
  - Production
- GENI Aggregate Manager in production
  - Allow provisioning of a sliver across AL2S as part of a larger GENI slice

Internet2 2015 Plans

- Continue to support network research on AL2S, and we are in particular interested in understanding and meeting needs of GENI researchers
- Deploy NSI on AL2S
  - Begin conversations about continued IDCP support
- ONOS Deployment, with Global Peers as a prototype service
- Work with vendors to get OF 1.3 Support
  - Brocade -> 5.8x (“now”) in testing
  - Juniper -> In discussion
- Continue to support and enhance OESS, FSFW
  - Evaluating OF 1.3 support in FSFW
- Refine Slice Deployment Process
  - Faster?
  - Test for correctness, then safety
  - Testing constraints?

Operating SDN Networks: The Good

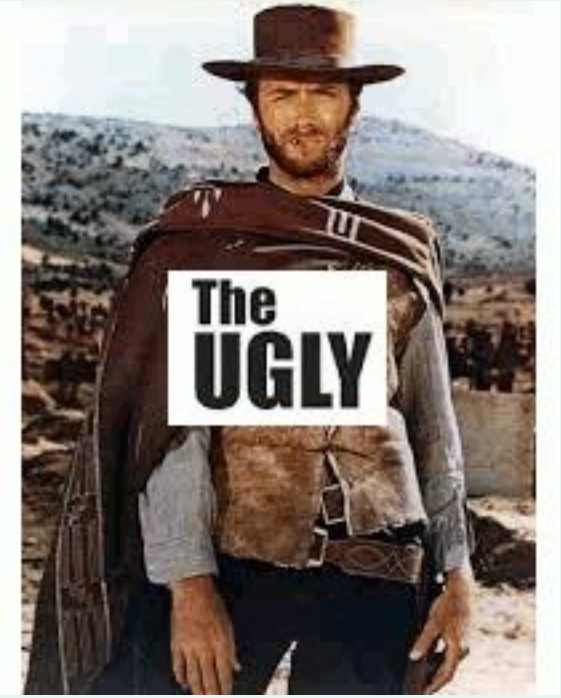
- Possible to build and operate a reliable Layer 2 and Layer 3 network atop an SDN Substrate
- Possible to support multiple controllers concurrently on an SDN substrate through network virtualization
- Possible to create a mult-domain SDX using network virtualization
- Possible to build a global Layer 3 network through software on a router-less network in ~1 month

Operating SDN Networks: The Bad

- Vendor implementations of OpenFlow 1.0 have been buggy and incomplete
- Vendor implementations of OpenFlow 1.3 have been very slow to appear, as well as buggy and incomplete
- OpenFlow 1.0 and 1.3 standards have too many optional features, making the implementation of new features a painstaking negotiating process with multiple vendors
- OpenFlow 1.0 and 1.3 specs are sufficiently “vague” that we had to write supplemental specs to ensure vendor interoperability

Operating SDN Networks: The Ugly

- Building a network software stack requires absolutely rigorous testing when *any* component changes
  - Testing harness becomes the resource bottleneck
  - Testing for safety != testing for correctness
- Supporting multiple controllers concurrently on a production network software stack:
  - Requires significant FTE resources
  - Moves slower than researchers are accustomed
  - Requires more productization (logging, release management, documentation) than normally done by researchers



Takeaways

- Operating an SDN-based network is doable, today, and has been for a 2+years
- SDN != Open SDN
  - SDN = Fully programmable devices
  - Open SDN = Fully programmable, vendor-swappable devices
- It's too soon to declare "winners" in the network stack space
  - Controllers: ODL, ONOS, Ryu, etc.
  - Apps: FSFW/OESS, SDN-IP, etc.
  - Declaring a "winner" raises the "narrow waist"

    => Less room for R&E innovation
- We need crisp, complete, required SDN programming interfaces fully implemented across multiple vendors
- We need to start tool development to support network operators of SDN-based-networks
- We need maturation of open source controllers
  - Logging, Documentation, Release Management, Long-Term Support
  - Open Source Testing Harnesses

JOHN HICKS

Network Research Engineer

SEPTEMBER, 2015

INNOVATION AND EXPERIMENTATION THROUGH SDN AND NETWORK VIRTUALIZATION

jhicks@internet2.edu

