ERIC BOYD

AND ED BALAS

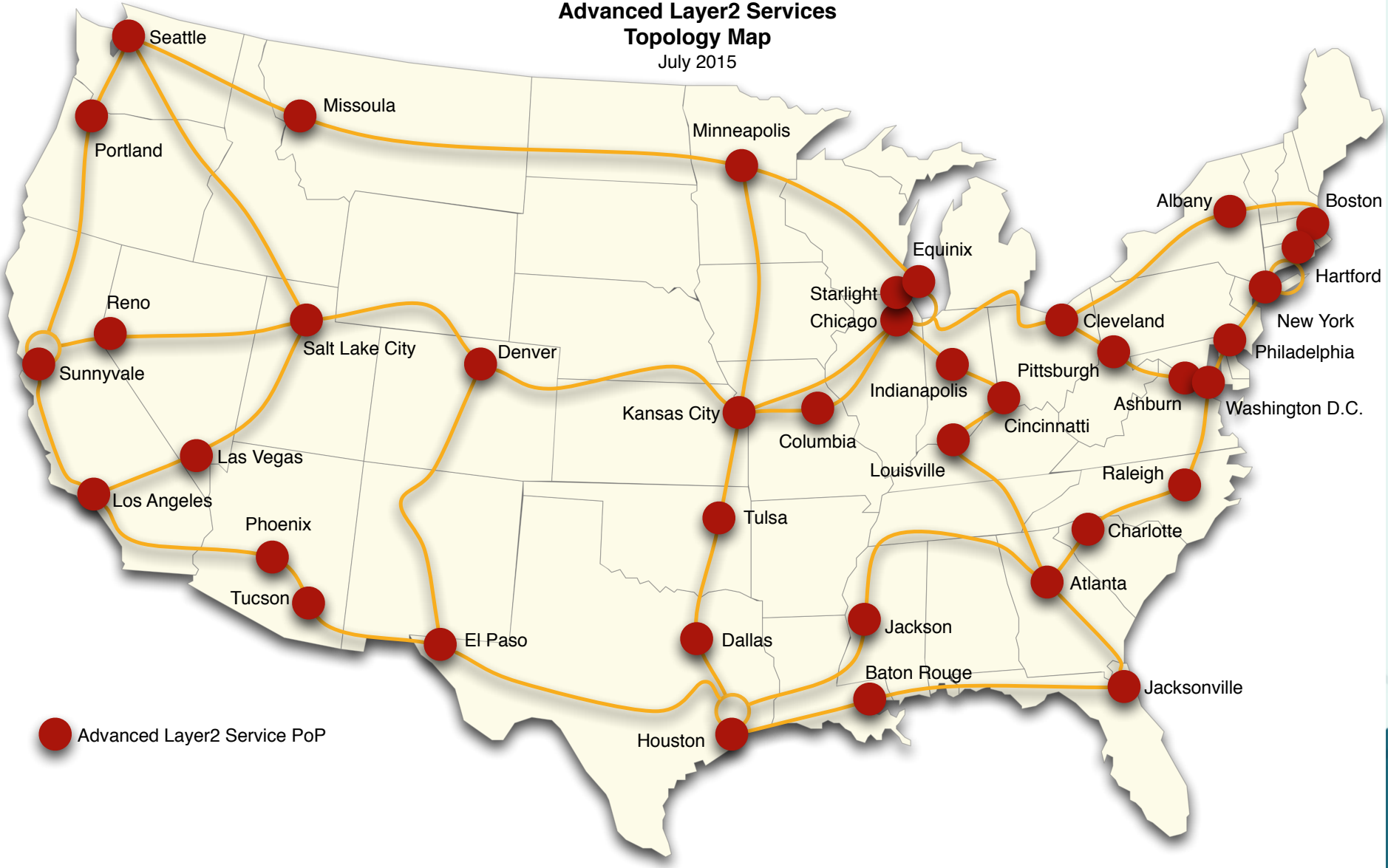
Internet2 and Indiana University

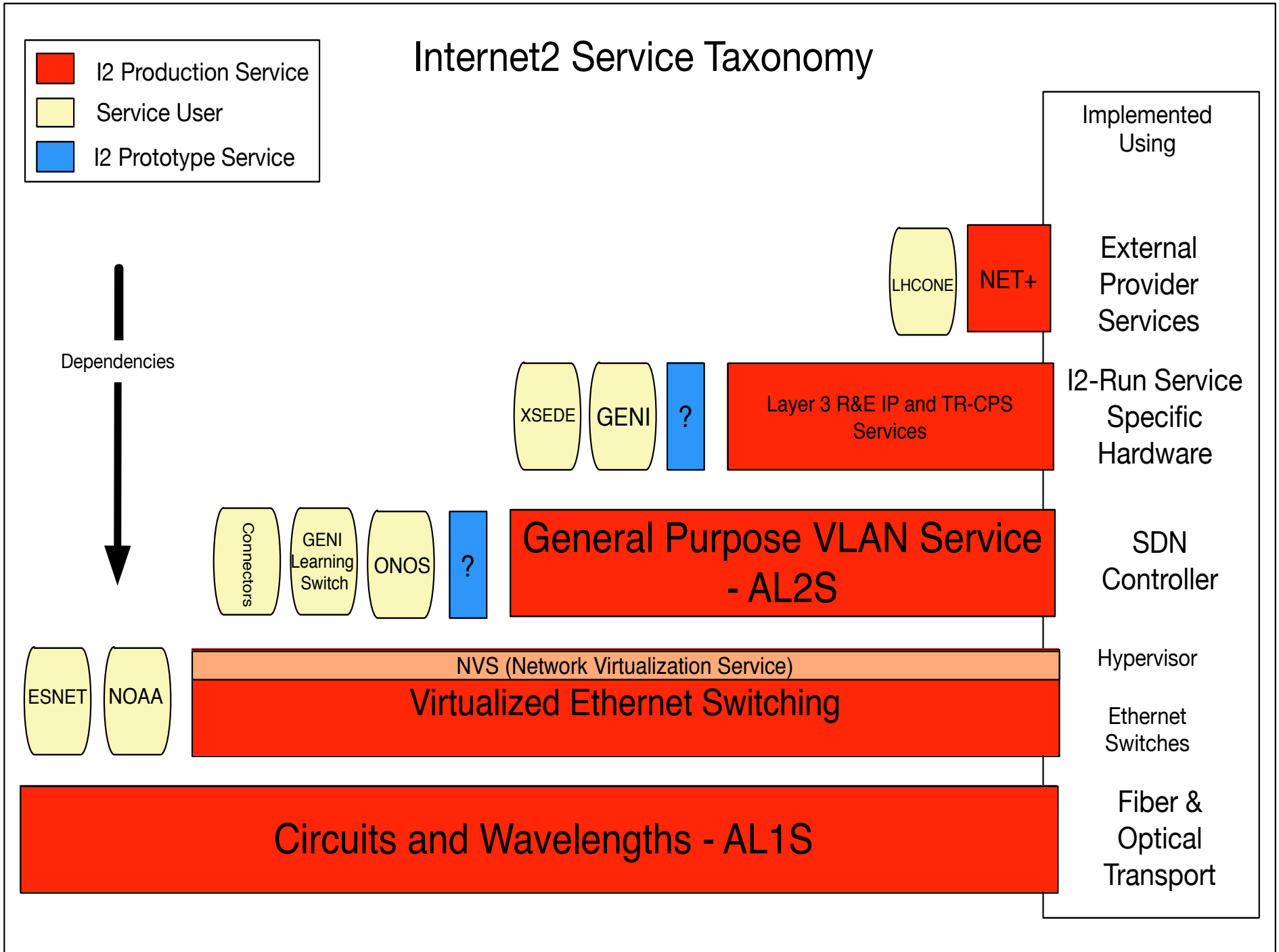
OCTOBER, 2015

Network Virtualization: A Retrospective After One Year

Internet2 Network

Advanced Layer2 Services
Topology Map
July 2015




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

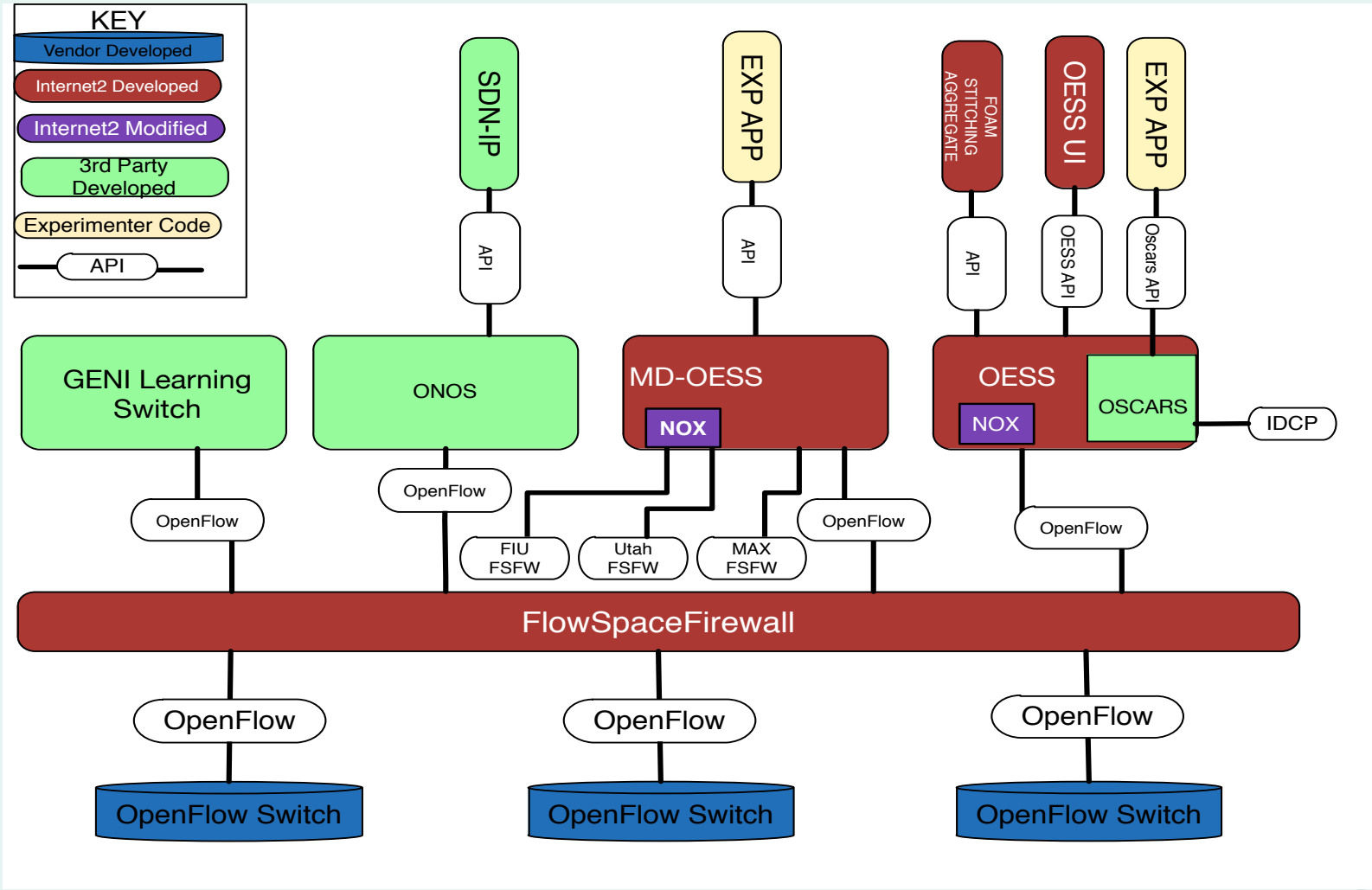
Notable Milestones to Date

- 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 Current Status: October, 2015

- OpenFlow 1.0 in production
  - OF 1.3 support in FSFW and OESS in design
  - Experimenting with Brocade 5.8bx implementation of OF1.3
  - Working with Juniper on implementing requirements for OF 1.3
- Hypervisor (FlowSpace Firewall1.0.6) in production
  - Supports L2 and L3 matching
  - Vendor Updates (current versions Juniper 13.3, Brocade 5.6dc)
  - Vendor-specific limits do exist.
- Controller (OESS 1.1.6a) in production
  - Supports Layer 2 Trace
- Accepting 3rd party controllers
  - Questionnaire
  - Openflow Network Emulator (Mininet clone of AL2S plus chaos monkey)
  - Test Lab
  - Production
- GENI Aggregate Manager in production
  - Allow provisioning of a sliver across AL2S as part of a larger GENI slice

Internet2 Network Software Stack



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 -> 5.8c (“now”) in testing
  - Juniper -> 15.1 in development
- 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

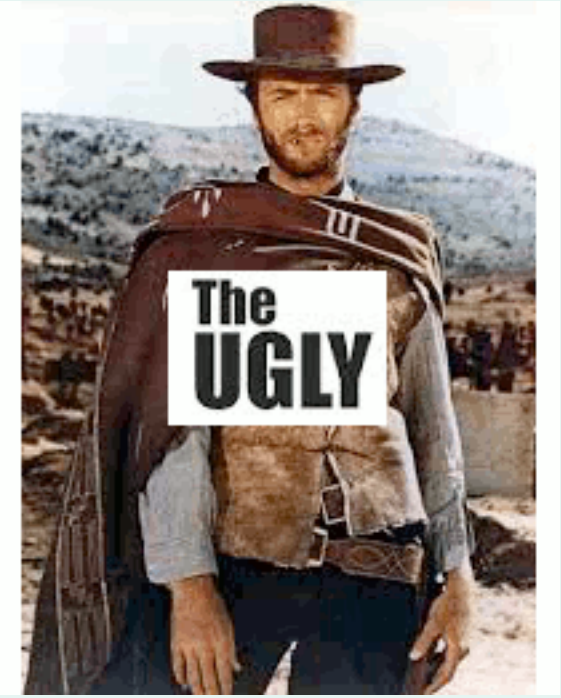
- Possible to build and operate a reliable Layer 2 and Layer 3 network atop an SDN Substrate
- Possible to support multiple controllers concurrently on an SDN substrate through network virtualization
- Possible to create a mult-domain SDX using network virtualization
- Possible to build a global Layer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



How we thought Slice testing would go

- App Developers fill out questionnaire
  - Assume testing against mininet
  - Assume systems best practice
- Developers demonstrate it working
- We go into lab testing
  - Assume it takes us a week to test
- Testing goes flawlessly
- New slice is recommended for deployment
- Deployment goes flawlessly
  - Assume lab tests have sufficient coverage
- Achievement unlock

What we saw in our first round

- Expectation mismatch on packaging slows testing and later deployment
- Testing was performed by developers with insufficient use cases
  - Found bugs in every component
  - Feedback loops with developers created protracted break, fix, test cycle
  - We found ourselves operating as a QA shop
- Getting to recommendation took several Months not a week
- In spite of testing , deployment still unstable
  - Scale based issues not protected by FSFW

Process Improvement

- Problems identified
  - App developers lacked test tools to mimic the complexity of production networks
  - Developers not anticipating the number of failures that will occur in a geo-distributed network
  - App developers were relying on us to help forcing us into their dev cycle basically.
- Adjustments we made
  - Build a test environment based on mininet and share with developers
    - Exact topo,
    - DPIDS and port names
    - production chaos
      - Link failures
      - Node reboots
  - Wait to test in our lab until they can test in virtual environment at “scale”

Take 2

- Developers initially skeptical
  - Felt the approach was overkill
- Developers changed tune
  - took a few months to resolve discovered issues
- Our lab testing was was wrapped in a week
- Deployment still unstable
  - We still have difficultly test at virtual scale with also involving vendor hardware
  - Insufficient vendor testing

Takeaways

- Operating an SDN-based network is doable, today, and has been for a 3 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

ERIC BOYD

AND ED BALAS

Internet2 and Indiana University

OCTOER, 2015

NETWORK VIRTUALIZATION: A RETROSPECTIVE AFTER ONE YEAR

eboyd@internet2.edu
ebalas@grnoc.iu.edu

