Seeing The Past, Present and Future: Macro Trends in Networking and the Role of Software Defined Networking

David Meyer
CTO and Chief Scientist, Brocade
Director, Advanced Technology Center, University of Oregon
TIP2013
Honolulu, HI

dmm@{brocade.com,uoregon.edu,1-4-5.net,...}
Agenda

• (Macro) Trends Inducing a New Landscape

• The Past: How We Got Here

• The Present: What Exactly is the Current State of Affairs?

• The Future: Where’s it All Going

• Q&A if we have time
Macro Trends
The Evolution of Intelligence
Precambrian (Reptilian) Brain to Neocortex \(\rightarrow\) Hardware to Software

- Shared Themes/Biological Metaphors
  - Thin-waist architectures
  - Massively distributed
  - Highly layered with Robust Control loops
  - Component Reuse

It's all about code!
BTW, there’s an apparent paradox

Component behavior *gratuitously* uncertain, yet systems have robust performance.

Darwinian evolution uses selection on random mutations to create complexity.

Network folks use what, exactly?
Everything De-silos

Vertical -> Horizontal Integration
Open {APIs, Protocols, Source}
Everything Pluggable
Future is about Ecosystems
Network Centric → IT Centric

- Shift in influence and speed
- Shift in locus of purchasing influence
- Changes in cost structures
  - ETSI NfV, ATIS, IETF, ...
- **NetOPs → DevOPs**
Other Important Macro Trends

• Everything Virtualizes
  – Well, we’ve seen this

• Data Center new “center” of the universe
  – Looks like ~ 40% of all traffic is currently sourced/sinked in a DC
  – Dominant service delivery point

• Integrated orchestration of almost everything

• Bottom Line: Increasing influence of software *everywhere*
  – All integrated with our compute, storage, identities, ...
The Past: How We Got Here

Basically, everything *networking* was to vertically integrated, tightly coupled, non-standard.

Goes without saying that this made the job of the network researcher almost impossible. So what happened?
In the Beginning...
(in)SANE
Too simple:
- Feature/functionality
- Expressiveness
The Present: Current (ONF) SOA

(a) Packets are matched against multiple tables in the pipeline

- Why this design? Combinatorics...
- But see also: IETF, ATIS, ETSI, ITU-T, MEF, ...
  - I2RS, ALTO, PCE, BGP-LS, ..
  - Different architectural model(s)
- Consider complexity: $\sim O(n! \cdot l^k)$ paths
- Emerging: SDN Continuum

Too Complex:
- Not implementable on ASIC h/w
- Breaks new reasoning systems
- No fixes for the lossy abstractions
- Architectural questions

Is the flow-based abstraction “right” for general network programmability?
A Simplified View of the SDN Continuum

Service Layers

Control and Orchestration
(overly simplified view)

Apps

Apps

Physical and Virtual Resources
(CSN)

OF/SDN
Properties:
-- Complete Separation of Control and Data Planes
-- Open Interface to the Forwarding Plane

CP/SDN
Properties:
-- Retains existing (distributed) Control Planes
-- Programmable control plane
-- Examples: PCE, IRS, vendor SDKs

OL/SDN
Properties:
-- Retains existing (simplified) Control Planes
-- Programmable overlay control plane
-- Examples: Various Overlay technologies

May be repeated
(stacked or recursive)
So The Future: Where’s it All Going?
But More Seriously....

- **High order bit:**
  - Cloudy crystal balls, architect for change and rapid evolution
  - “agility”
  - Increasing roles for s/w and programmability

- **Conventional Technology Curves – S & F**
  - Moore’s Law and the reptilian brain
    - Someone eventually has to forward packets on the wire
  - 400G and 1T in the “near” term
  - Silicon optics, denser core count, ....

- **Ecosystems**
  - Open Interfaces: Protocols, APIs, Code, Tool Chains
  - Open Control Platforms at every level
  - “Best of Breed” markets
  - *Recursive Programmable Network Stacks*

- **BTW, open source/open source consortia dominate**
  - And what is the role of standards bodies in age of Open Source?
Programmable Network Stack Cartoon

- Virtual and Physical Forwarding Resources, Compute and Storage
- Overlays, VPNS, Network Slicing
- Distributed Routing and Peering
- APIs and Protocols
- Cloud/Tenant Orchestration, Services, Management
- APIs and Protocols
- Services Layer (GOTOM, IM/Presence, Video, Mobility, …)
- Recursive
Finally: A Cautionary Tale
The More Things Change...
(Dave Clark, IETF 24, 1992)

As the Internet and its community grows, how do we manage the process of change and growth?

- Open process – let all voices be heard.
- Closed process – make progress.
- Quick process – keep up with reality.
- Slow process – leave time to think
- Market driven process – the future is commercial.
- Scaling driven process – the future is the Internet.

We reject: kings, presidents and voting.
We believe in: rough consensus and running code.

What are we good at?
- Responding to short term reality.
- Building stuff that works.
- Calling bad stuff bad.

What are we bad at?
- Growing our processes to match our size.
- Setting long-term direction.

Software
Open Source
Q&A

Thanks!