The evolving Internet industry

Disaggregation, SDN and NFV
Coming up later this week....

**Wednesday**: Jeff Mogul (Google)
He will build on some of the things you will learn today.

**Friday**: (Nick) Network Virtualization.
The Design Philosophy of the DARPA Internet Protocols

David D. Clark
Massachusetts Institute of Technology
Laboratory for Computer Science
Cambridge, MA. 02139

(Originally published in Proc. SIGCOMM ’88, Computer Communication Review Vol. 18, No. 4, pp. 106–114)

Abstract

The Internet protocol suite, TCP/IP, was first proposed fifteen years ago. It was developed by the Defense Advanced Research Projects Agency (DARPA), and has been used widely in military and commercial systems. While there have been papers and specifications that describe how the protocols work, it is sometimes difficult to deduce from these the design of the protocol in another group. The Internet architecture is now split into the IP and TCP layers. This seems basic to the design, but was also not a part of the original proposal. These changes in the Internet design arose through the repeated pattern of implementation and testing that occurred before the standards were set.

The Internet architecture is still evolving to meet new uses.
Architectural Principles

1. Dumb and simple
   - Streamlined, fast, low-cost, easy to maintain, infrequent upgrades.

2. Reliable in-order delivery can be built on top, at the end-points.
   - Intelligent end-points (computers).
Architectural Principles

3 Decentralized control
→ Rapid organic growth

The number of Internet users in the world

Source: http://www.internetlivestats.com/
Tension

The Internet was carefully designed ...

• To be simple and streamlined
• To have decentralized control: Lots of individually controlled pieces.

Which led to ...

• Explosive organic growth of the Internet.
• A great business for companies selling routers.
What public Internet routers do
function Dijkstra(Graph, source):

for each vertex v in Graph:
    dist[v] := infinity ;
    previous[v] := undefined;

dist[source] := 0 ;

Q := the set of all nodes in Graph ;

while Q is not empty:                 // The main loop
    u := vertex in Q with smallest distance in dist[] ;
    remove u from Q ;
    if dist[u] = infinity:
        break ;

    for each neighbor v of u:
        alt := dist[u] + dist_between(u, v) ;
        if alt < dist[v]:
            dist[v] := alt ;
            previous[v] := u ;
            decrease-key v in Q;

return dist[], previous[];
end function

Edsger Dijkstra
1930-2002
"If a packet is going to B, then send it to output 3."

1. Figure out which routers and links are present.
2. Run Dijkstra’s algorithm to find shortest paths.
1. Figure out which routers and links are present.
2. Run Dijkstra’s algorithm to find shortest paths.
1. Figure out which routers and links are present.
2. Run Dijkstra’s algorithm to find shortest paths.

Building a reliable distributed system is really hard!
Number of published Internet RFCs


Number: 0, 1,000, 2,000, 3,000, 4,000, 5,000, 6,000, 7,000
By 2005

- Internet routers were complex: Over 100M lines of source code
- They were closed and proprietary, vertically integrated, expensive.
- Very hard to manage or control as a network. No clean APIs.
- ISPs felt under a “stranglehold” by the router manufacturers
- Internet research community was frustrated:
  - Internet was “ossified”
  - Easy to invent new ideas; but hard to test and deploy
Two parallel trends

1. **Research community**
   - New programs: Clean Slate Program, NSF programs
   - New research: Questioning closed ecosystem
   - New switch API: OpenFlow
   - New open source: Control planes, switches, emulators

2. **Networking industry**
   - Rise of data centers
     - Rise of Linux
     - Availability of merchant switch chips
     - Disaggregation of servers
   - ISPs under stress
     - Falling consumer prices
     - Need to differentiate

Somehow, control needed to move from the equipment vendors to those who own and operate networks
Example: Data Center Owner

Cost
- 500,000 servers
- 25,000 switches
- $10k per switch = $250M
- $2k commodity switch = $50M
- Savings in 5 data centers = $1B

Control
- More flexible control of traffic
- Tailor network as needed
- Quickly improve and innovate
- e.g. Google B4 network from 50% to 95%
Example: Internet Service Providers

Global IP traffic growing 50% per year
End-customer monthly bills stay the same
Therefore, cost of ownership needs to reduce 50% per Gb/s per year
But in practice, reduces by <20% per year
Research community in 2006

1. Stanford Clean Slate Program
   “What would we do if we started (the Internet) again?”

2. Martin Casado and the Ethane Project
How difficult is it to define all network operations in software, outside the datapath?

Stanford campus

2006

35,000 users
10,000 new flows/sec
137 network policies

2,000 switches
2,000 switch CPUs
Crazy question: What if software decides whether to accept each flow, and how to route it?
How many $400 servers do we need for 35,000 Stanford users?

Answer: Less than one!
If we can define network behavior outside the datapath, then eventually we will.
Software Defined Network (SDN)

Control Program

Global Network Map

Control Plane

Control Forwarding

Control Forwarding

Control Forwarding

Control Forwarding

Control Program

Control Program

Control Program

OpenFlow:
Dijkstra, IS-IS, BGP, MPLS, Firewall...

Global Network Map

Network OS

Forwarding

Forwarding

Forwarding

Forwarding
Quickly led to...

OpenFlow, SDN, Open vSwitch, network virtualization, ...

In 2005, every data center and ISP was built using closed, proprietary routers.

Today, the top 10 data center owners all build their own, and write their own software (or use open source)
Computer Industry

- Specialized Applications
- Specialized Operating System
- Specialized Hardware

Open Interface: Windows (OS) or Linux or Mac OS

Microprocessor
Networking Industry

Specialized Features
Specialized Operating System
Specialized Hardware

Open Interface
NOX Beacon ONIX POX ONOS Floodlight Trema ODL Ryu
Switch Chips
Network Function Virtualization (NFV)

Packet Forwarding

Middlebox

Packet Forwarding

Middlebox

Packet Forwarding

Middlebox

Packet Forwarding

Public Internet

Firewalls
Load-balancing
NAT
Boundary routers
Deep Packet Inspection
DDoS Mitigation
Network Function Virtualization (NFV)

VMs connected to Packet Forwarding nodes, which in turn are connected to the public internet. Key functions include:
- **Firewalls**
- **Load-balancing**
- **NAT**
- **Boundary routers**
- **Deep Packet Inspection**
- **DDoS Mitigation**
With hindsight, Disaggregation, SDN and NFV were inevitable

Part of a bigger trend towards the owners and operators of networks taking control of how they work
Inevitable because...

1. Rise of Linux.
2. Rise of baremetal servers and data centers.
3. SDN: Rise of merchant switching silicon.
4. NFV: Rise of computer virtualization.
“The Future of Networking and the Past of Protocols”
Scott Shenker 2011
Coming up later this week....

**Wednesday:** Jeff Mogul (Google)
He will build on some of the things you will learn today.

**Friday:** (Nick) Network Virtualization.