openSDN: A Service Delivery Network Architecture for Future Internet Evolution

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu
Talk at HP, Palo Alto, May 12, 2011
Audio/Video Recordings of this talk are available at
http://www.cse.wustl.edu/~jain/talks/sdn_hp.htm
Overview

1. Five Trends in Networking
2. Five Key Features that Services Need
3. Five Architecture Design Principles for Success
4. Five Key Components of Architecture
5. Five Features of OpenSDN
Why 5?

- It’s a Fermat’s number, Fibonacci Number, Pell Number, Markov Number, Catalan Number, Smallest twin prime, Safe Prime, Mersenne Prime, Factorial Prime, Eisenstein Prime, Wilson Prime, …
- Don’t want to bore you with more than 5 points
Five Trends in Networking

1. Moore’s Law
2. User Multihoming + Mobility
3. Wireless Edge
4. Declining Revenues in Transport
5. Profusion of Services
Trend 1: Moore’s Law

- Computing Hardware is cheap
- Memory is plenty

⇒ Storage and computing (Intelligence) in the net

- Energy
- Space
- Communication in Space
- Link

- Matter
- Time
- Communication in Time
- Storage (USB, Caching,…)

Next Gen nets will use storage in networks, e.g., DTN, CCN
Trend 2: Multihoming + Mobility

- Centralized storage of info
- Anytime Anywhere computing
- Dynamically changing Locator
- User/Data/Host/Site/AS Multihoming
- User/Data/Host/Site Mobility
  ⇒ ID/Locator Split

Mobile Telephony already distinguishes ID vs. Locator
We need to bring this technology to IP.
Trend 3: Wireless Edge

1. Billions ⇒ Scalable
2. Heterogeneous ⇒ Customization of content
3. Slow ⇒ Bottleneck ⇒ Receiver Control
   (IP provides sender controls but no receiver controls)

Need to design from receiver’s point of view
Trend 4: Declining Revenues in Transport

- Telecom carriers' disappearing revenues in basic transport
- New opportunities in apps and Intelligent transport

Future of ISPs is to go beyond best effort trucking services
Trend 5: Profusion of Services

- Almost all top 50 Internet sites are services [Alexa]
- Smart Phones: iPhone, Android Apps
  ⇒ New globally distributed services, Games, …
  ⇒ More clouds, …

Networks need to support efficient service setup and delivery

Washington University in St. Louis http://www.cse.wustl.edu/~jain/talks/sdn_hp.htm ©2011 Raj Jain
Service Center Evolution

Single Server

Load Balancer

Multi-Server

Distributed Services

Global Internet

Need a distributed load balancer for globally distributed datacenters
Globally Distributed Services

- Scale ⇒ Global ⇒ Distributed ⇒ Multihomed
- Internet 1.0 is designed for point-to-point communication
- Significant opportunities for improvement for global services
Globally Distributed Services (Cont)

- It’s the service responsibility to find the right server for the client

Google.us  Google.in  Google.cn

Internet
Trend: Private Smart WANs

- Services totally avoid the Internet core ⇒ Many private WANs
- Google WAN, Akamai ⇒ Rules about how to connect users

Diagram:

- Google Data Center
- Google Data Center
- Google Data Center
- Google’s WAN
- Internet
- Access ISP
- Access ISP

Opportunity for ISPs to offer these types of WAN services
OpenSDN

- High-Speed WAN architected for Service Delivery.
- Allows ASPs to quickly setup services
Ten Key Features that Services Need

1. **Replication**: Multiple datacenters appear as one
2. **Fault Tolerance**: Connect to B if A is down
3. **Load Balancing**: 50% to A, 50% to B
4. **Traffic Engineering**: 80% on Path A, 20% on Path B
5. **Flow based forwarding**: Movies, Storage Backup, …
   - ATMoMPLS, TDMoMPLS, FRoMPLS, EoMPLS, …
   - Packets in Access, Flows in Core
6. **Security**: Provenance, Authentication, Privacy, …
7. **User Mobility**: Gaming/Video/… should not stop as the user moves
8. **Service composition**: Services using other services
9. **Customization**: Every service has different needs
10. **Dynamic Setup** ⇒ Networking as a Service
Five Arch Design Principles for Success

1. Evolution not replacement
2. Coexistence (Backward compatibility)
3. Incremental Deployment
4. Economic Incentive for first adopters
5. Customization without losing control
Networking: Failures vs Successes

- 1986: MAP/TOP (vs Ethernet)
- 1988: OSI (vs TCP/IP)
- 1991: DQDB
- 1994: CMIP (vs SNMP)
- 1995: FDDI (vs Ethernet)
- 1996: 100BASE-VG or AnyLan (vs Ethernet)
- 1997: ATM to Desktop (vs Ethernet)
- 1998: ATM Switches (vs IP routers)
- 1998: MPOA (vs MPLS)
- 1999: Token Rings (vs Ethernet)
- 2003: HomeRF (vs WiFi)
- 2007: Resilient Packet Ring (vs Carrier Ethernet)
- IntServ, DiffServ, …

Technology alone does not mean success.
Five Architecture Design Principles

1. Evolution not replacement.
2. Coexistence (Backward compatibility): Old on New. New on Old
3. Incremental Deployment
4. Economic Incentive for first adopters
5. Customization without loosing control (No active networks)

Most versions of Ethernet followed these principles. Many versions of IP did not.
Five Key Components of Architecture

1. Naming
2. Data Plane (Forwarding)
3. Control Plane (Routing)
4. Management Plane (Monitoring, Fault tolerance, …)
5. Security
OpenSDN Features Overview

1. Attribute Based Naming
2. Separation of Control and Data Plane
3. Rule based delegation
4. Strong Security
5. Packet and flow based communications
Naming

- Globally unique name with attributes
  ⇒ Attribute based naming
- Attributes: Location, Type
- IDs: Service ID, Host ID, Data ID, User ID, Infrastructure Point-of-Attachment ID (= Locator)
- Applications are bound to IDs
- All IDs are 128-bit
  ⇒ No changes to current applications
Rule Based Delegation

- Control Interface: Registration of Rules
  ⇒ Customization

- Data Interface: Enforcement of Rules

Control Plane

Rules

Data Plane
(Rule enforcement)
Security

- Control Plane Security: Rules Registration, Distribution, Updates
- Data Plane Security: Provenance, Authentication
The Narrow Waist

- Everything as a service over service delivery narrow waist
- IP, HTTP, Content, Service delivery, …
SDN Applications

1. Telecom Services
2. Critical Infrastructures
3. Private WANs
4. Scientific Computing
5. Datacenters

SDN
Application 1: Telecom Services

- IP Multimedia, Video Conferencing, Gaming, …

Computing resources dynamically optimized on demand

Networking resources dynamically optimized on demand
Application 2: Critical Infrastructure

- Defense, Power Grid, Water supply, Gas Supply, ...
- Security + Customization
  ⇒ Multiple services can share a single SDN
Application 3: Private WANs

- Multiple sites (including cloud computing) with rules for traffic handling
Application 4: Scientific Computing

- Distributed computing using high-speed networking,
- National Knowledge Network
Application 5: Datacenter

- Multiple services and clients in a datacenter
- SDN design is good for short distance too
Implementation: OpenSDN Appliance

- Service A1
- Service B1
- Service A2
- Service b2

Internet

Legacy Router
openSDN Appliance

Access ISP

End User Hosts

Access ISP

End User Hosts
Summary

1. Profusion of services on the Internet
2. OpenSDN is an overlay designed for service delivery
3. New architectures need evolution, backward compatibility, incremental deployment, economic incentives, customization without losing control for success
4. Services need replication, fault tolerance, traffic engineering, security, …
5. OpenSDN provides these features with rule-based delegation, support for legacy nodes, data-control plane separation

Service Delivery: Opportunity for ISP’s and equipment vendors