John Mailhot
Imagine Communications

Transition to IP-based Signals in the Facility
400.0 Gbits

100.0 Gbits

10.0 Gbits

1.0 Gbit

0.0 Gbit

0.270 G

1.5 Gbit

3.0 Gbit
Will IP [for video] Replace Coaxial Cable?

256x256 SDI(3G) Router
Fully Non-Blocking
768 Gbits/sec throughput (3G SDI)

COTS Ethernet Switch/Router
36x40GBE in 1RU (full duplex)
equivalent of 144x10G ports
1RU Fully Non-Blocking
1440 Gbits/sec throughput
Moore’s Law is not the Whole Story -- A tale of Three Economies

- The “Broadcast Equipment” Economic Ecosystem
- The “Telecom Infrastructure” Economic Ecosystem
- The “Information Technology” Economic Ecosystem

Standard Platforms
Virtual Machines
IP/Ethernet Networks
How does that play out in practice?

Dual 10 GBE NIC
$270

PC HDSDI I/O
~$1000

HDSDI I/O For Audio Console
$4399
Why is Now the Right Time to Transition to IP?

- **Perfect Storm of Four Factors**
  - Declining cost/Increasing Capability of Commercial IT and IP Equipment
  - 10GBE has achieved market reality and scale → HD over IP at reasonable cost
  - IT-based workflow technologies and tools have matured and are mainstream
  - Enterprise-Class “Cloud” (Private and Public) Flexibility Model

- **The Transition is Starting to Affect Facility Designs Worldwide**
  - Economics of Commercial IT infrastructure get better every year
  - Cross-over point depends on the type of workflow needs
  - Every part of the facility is moving to the COTS Infrastructure
IT-Industry COTS Equipment

- **COTS – Commercial Off-The-Shelf**
  - High Volume IT marketplace drives “Moore’s Law” cost and price trends

- **Ethernet Switches with Hundreds of 10GBE ports each**
  - Cisco, Brocade, Arista, Extreme, Juniper, and 10 smaller vendors
  - Cost-per-port for 10GBE is going down rapidly – no end in sight

- **High-Density Computing Platforms**
  - HP, Cisco, IBM, Dell, SuperMicro, and 10 smaller vendors
  - Cost-per-Core going down rapidly – no end in sight
SDI → IP is Part 2 of a 3-Part Story

1. Better CPUs → Servers, Graphics, Editors became 100% Software-based Devices (except for I/O)

2. Better Ethernet → Pixels-over-IP becomes cheaper than SDI-based interfaces (managed through SDN techniques)

3. Better Systemization → COTS Network Equipment and COTS Computing Servers enable pure *Software-Defined Workflows*
What is Software-Defined Networking (SDN)?

- A Modern Approach to IT Networking which *Separates the CONTROL* of the traffic *away from the switching* devices (data plane)
- Enables *Application-Specific knowledge* to decide which signals go where, instead of relying on the COTS switches to decide
- Enables use of Commercial Datacenter-Grade Switches for Broadcast Video
What Problem is SDN trying to solve?

- IP Networks work GREAT – every office has at least one
  - When you use it in a simple, standard way, with Gateway Routing and NAT

- But we’ve always had some challenges with IP networks, when we want more
  - Where is the network sending my packets? Why did the network do what it did?
  - What path is my data taking right now? **How do I make it do what I wanted?**

- But virtualized datacenters bring even bigger challenges
  - If I move this virtual machine from one server to another, how do I get all the right network connections to follow it?
  - Can I write a script (triggered by a web services engine) that turns up a VM and creates all the network connections to/from it?
Researchers at UC Berkeley and Stanford wanted to do research on new network protocols and routing algorithms
• But they couldn’t make the switches they had do what they wanted
• What they really wanted was to
  – turn off the intelligence “in” the switches, and
  – have their own software program the forwarding tables, then
  – have the switches act on the tables they made

This grew into the “OpenFlow” whitepaper in March, 2008
  – Co-Authors from Stanford, MIT, UC Berkeley, Washington University, and Princeton

This also marked the beginning of the OpenFlow consortium to develop and maintain the standards and coordinate interoperability tests
What did the OpenFlow paper propose?

- Separate switch ports into “normal” ports and “openflow” ports.
- Use a secure channel to connect an external controller (with research software) to the standard switch.
- Provide a (vendor independent) control schema for the external controller to program the flow table matching engine of the openflow ports.

- More bluntly: Separate the **Smart Management** away from the **high-volume packet processing** through a common (secure) interface.
And then what happened?

- The researchers were happy, and went on to do good research

- The Switching Industry business dynamics got more “complicated”
  - Fancy Routing Protocol Processing is a differentiator among vendors
  - New “white-box” switches emerged that do nothing except OpenFlow
  - Switch Makers talk about supporting OpenFlow on more models soon

- The DataCenter Industry uses these techniques to provide virtual hosting services, virtual machine redundancy and clustering, and virtual private clouds

- Large Data Enterprises (Google, Amazon, PayPal, etc) use it daily
OpenFlow is really one small part of SDN

- Network Configuration Driven by High-Level Application
- SDN Controller has visibility of the whole network
- Multiple switches, geo-diverse, with common control structures
Assuming I’m not writing it, where does that “SDN Control Software” Come From?

- Many, Many Open Source Projects:
  - Beacon, Floodlight [Floodlight is fairly popular]
  - POX, IRIS, MUL
  - Loads of others that came and went

- Switch Vendor Offerings:
  - Each vendor has some system that offers SDN-like functionality packaged as a commercial offering (generally leveraging proprietary features)

- Open Daylight Project
  - A Funded-Multi-Vendor-Collaboration – Producing open source
What is OpenStack?

- Networking is one part of a larger business challenge: How to operate a multi-tenant hosting datacenter with high reliability, application-level redundancy, and scale?

- The datacenter industry (led by rackspace.com) funded an opensource project to develop solutions for this larger problem
  - [http://www.openstack.org/](http://www.openstack.org/)

- OpenStack has several inter-related software projects
  - Computing, Storage, Networking, Orchestration, Dashboard, Identity, etc.
  - Over 180 companies contributing time and effort
What can you Do with OpenStack?

- Stand up your own “private Cloud” and host applications on it
  - Manage multiple tenants, with quotas and permissions
  - Arrange Networking between the applications, and storage for the apps

- Run applications in one of several “public clouds” like AWS

- Monitor and Manage the running hosted applications through a dashboard system
What Does Any of This Have to do with Video?

256x256 SDI(3G) Router
Fully Non-Blocking
768 Gbits/sec throughput (3G SDI)

COTS Ethernet Switch/Router
36x40GBE in 1RU (full duplex)
equivalent of 144x10G ports
1RU Fully Non-Blocking
1440 Gbits/sec throughput
Signal Topologies of the Future Plant

- SDI Router in the core, with a few IP things translated back to SDI
  - Low-risk, add IP only as devices natively speak it
  - Requires translation where IP meets baseband
  - Difficult endgame – you still have a baseband core

- IP Switches in the middle, everything converged to IP
  - Best-cost approach as more and more devices natively speak IP
  - Works well with top-of-rack architecture for signal management
  - Requires translation for baseband devices to get to IP
  - Wire-level redundancy available for maintenance and uptime

- Hybrid Core – Baseband plus IP
  - Best of Both Worlds – baseband for baseband, IP for IP
  - Translators as tie-lines – automatically allocated and routed
  - Most cost-effective in the “Middle zone” of the crossover
  - Clear path to the all-IP endgame
Industry Timing Models for Flow Switching

- **Source-Timed:** delineate frame sequence in a matchable header field
  - Pre-set routes based on matching this field
  - Requires tight alignment (timing) between sources much like SDI

- **Switch-Timed:** use logic inside the switch (like FPGA-enabled logic) to do the switch-over inside the switch. Requires timed sources and special switches

- **Destination-Timed:** Destination begins receiving new flow before leaving the old one, and destination performs the alignment of the two flows.
Saying “Use IP” is like Saying “Use Coax”

- Coaxial Cable:
  - Composite, L-Band, IF, SDI, ASI, RF, ...

- IP/Ethernet:
  - Pixels, Compressed Pixels, FTP, Controls, PTP, ...
  - Email, Web Pages, ...
What is the “right thing” to put over IP?

- **Highly Compressed Signals**
  - Typical for MPVDs [Cable Head-Ends, DTH Satellite]

- **Mezzanine Compressed Signals**
  - Useful for trading off bandwidth cost and equip cost

- **Uncompressed (Pixels) over IP**
  - Lowest software footprint
  - No discussion about picture quality
Standards, Standards, Standards

- There is a lot of work going on in a lot of committees to define this all
  - VSF/SMPTE/EBU JT-NM
  - IETF
  - ONF, ODL and related projects
So What’s Stopping Us?

- SDI is still very cost-effective for many things

- The economics of “going IP” are best when the equipment in the plant natively speaks IP

- As the endpoints evolve IP interfaces, the core becomes IP
Thank You

John Mailhot
System Architect for IP Convergence
Imagine Communications