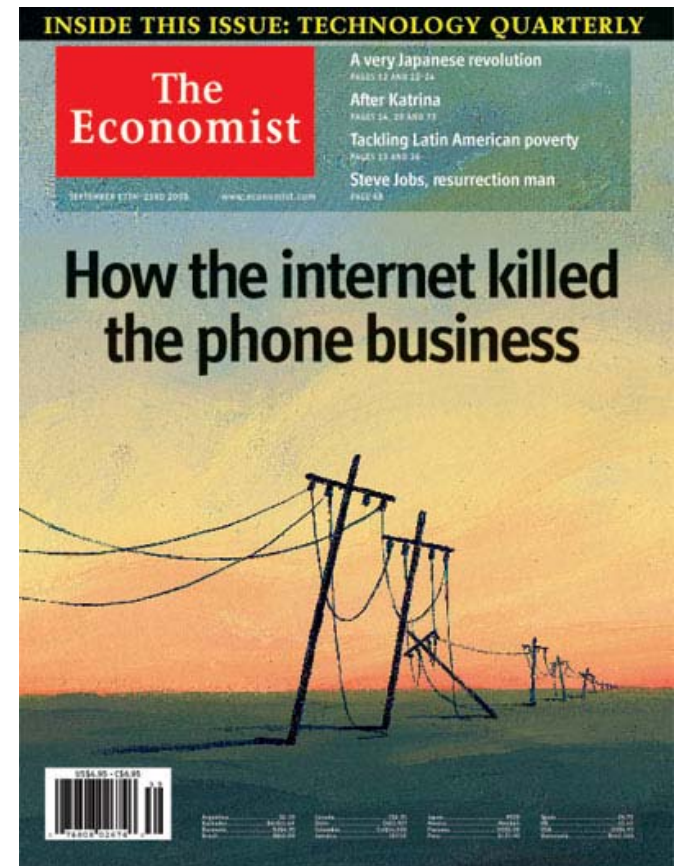


# **Cabo: Concurrent Architectures are Better than One**

Nick Feamster, Georgia Tech  
Lixin Gao, UMass Amherst  
Jennifer Rexford, Princeton

# Internet Faces New Demands

- High availability and performance
  - Path diversity, low-latency paths, fast reaction to failures, convergence...
- Security guarantees
  - Defense against unwanted traffic
- Manageability
  - Fault detection and localization
- Scaling
- Mobility



# New Protocols to the Rescue

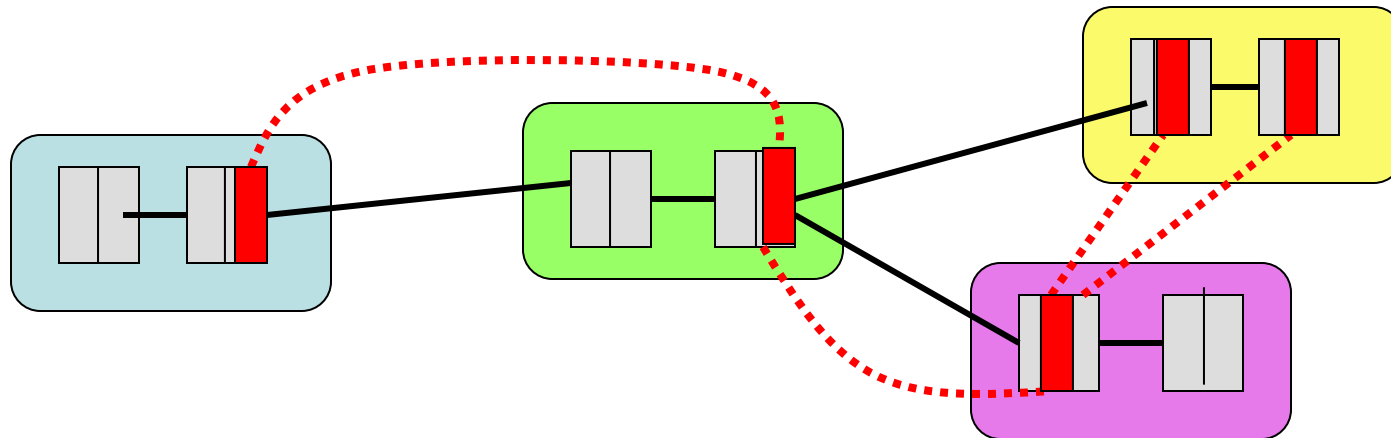
- **Addressing:** IPv6, 8+8, HIP, NIRA
- **Security:** Secure BGP (S-BGP), soBGP, SPV
- **Routing:** HLP, RCP, Compact/Valiant Routing
- **Naming:** DOA
- **Unwanted traffic:** Capabilities, SANE, DoS-Resistant Internet Architecture,

# The Coordination Constraint

- **Key problem:** Federation
  - No Internet Service Provider has control over an entire end-to-end path
  - This makes deployment, troubleshooting, accountability, etc. very difficult
- **Idea:** Make the infrastructure support multiple architectures in parallel
  - Separate providers of infrastructure and service

# Concurrent Architectures are Better than One (“Cabo”)

- **Infrastructure:** physical infrastructure needed to build networks
- **Service:** “slices” of physical infrastructure from one or more providers



**The *same entity* may sometimes play these two roles.**

# Network Virtualization: Characteristics

## Sharing

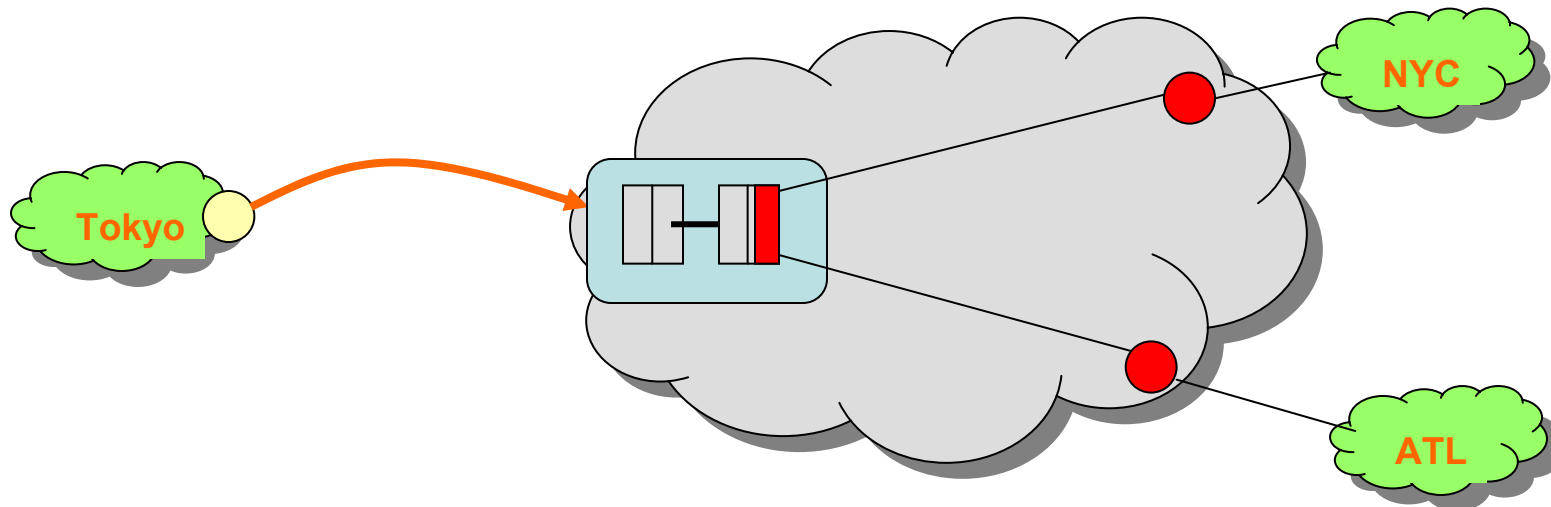
- Multiple logical routers on a single platform
- Resource isolation in CPU, memory, bandwidth, forwarding tables, ...

## Customizability

- Customizable routing and forwarding software
- General-purpose CPUs for the control plane
- Network processors and FPGAs for data plane

# Application: Virtual Co-Location

- **Problem:** ISP/Enterprise wants presence in some physical location, but doesn't have equipment there.




- **Today:** Backhaul, or L3 VPN from single ISP
- **Cabo:** Lease a slice of another's routers, links

# Application: End-to-End Services

- **Today:** Deployment logjam
  - Deployment requires consensus and coordination
- **Instead:** Adopt pluralist approach
  - Determined service provider leases infrastructure and deploys technology end-to-end

**Example**

*More Security* ←  *More Complete Reachability*

	Online Banking	Web Surfing
Routing	Secure routing protocol (e.g., S-BGP)	Lowest common denominator
Addressing	Self-certifying addresses (optimized for persistence)	Dynamic addresses (optimized for convenience)

# Application-Specific Networks

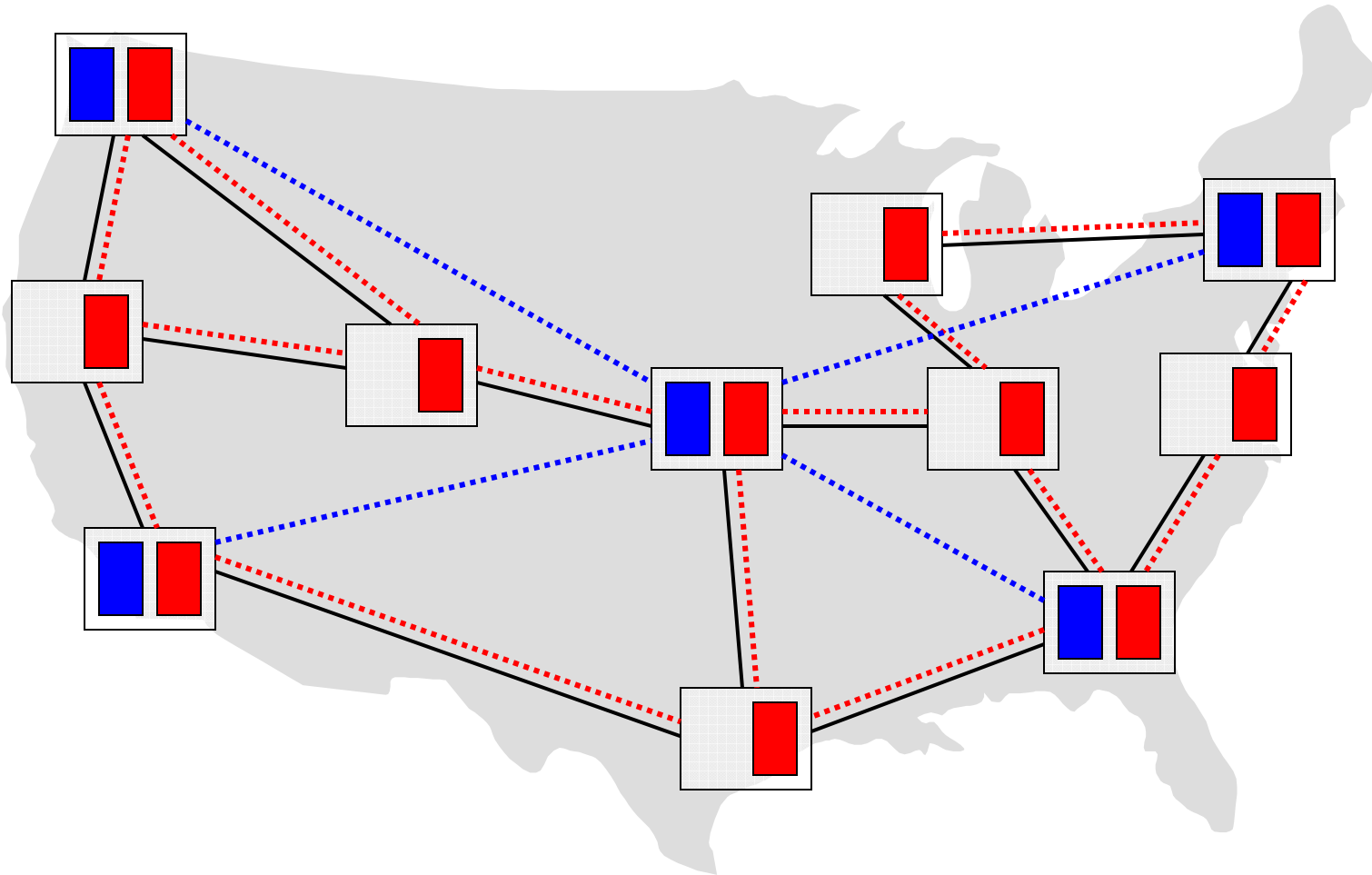
- **Today:** Optimize a single set of protocols
- **Instead:** Parallel deployment
  - Run multiple networks, each catered to specific applications

	<i>Better Scalability</i>	<i>Faster Convergence</i>
<b>Example</b>	Internal BGP	Link-State Protocols
Dissemination	Hierarchical, incremental	Flooding
Computation	BGP-style decision process	Shortest Paths

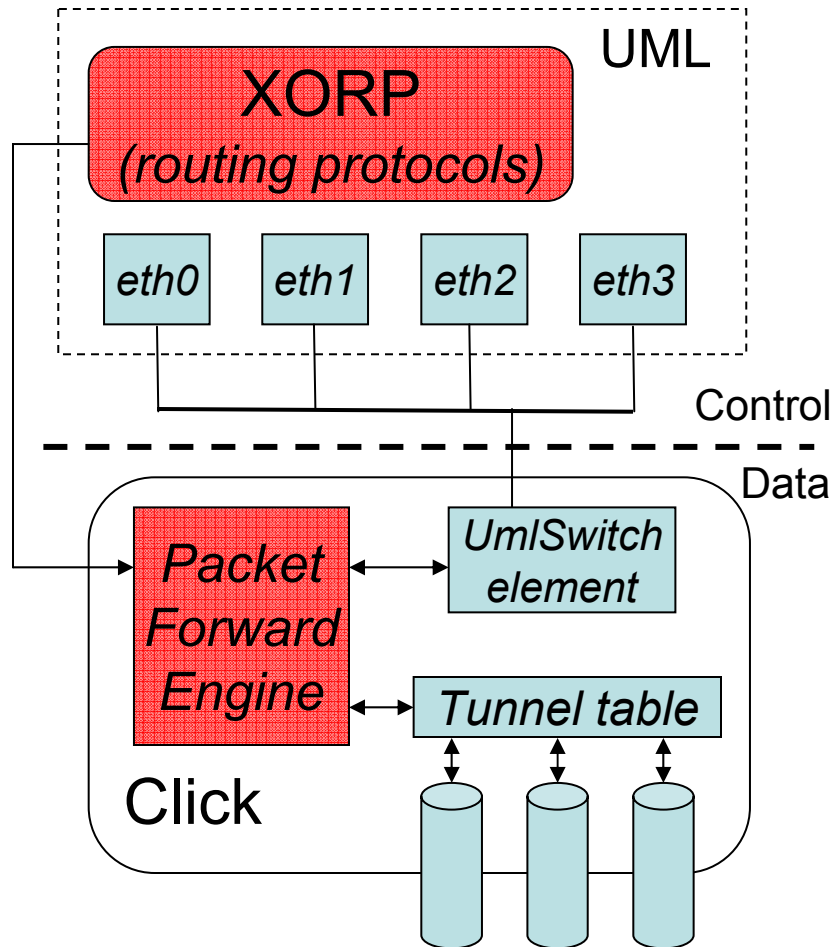
# Challenge #1: Simultaneous Operation

- **Problem:** Service providers must share infrastructure
- **Approach:** Virtualize the infrastructure
  - Nodes (lessons from PlanetLab will help)
  - Links (previous lessons in QoS?)
- VINI: Virtual Network Infrastructure
  - Cabo will exploit many of the same functions that are needed for VINI
  - Cabo philosophy: virtualization *is* the architecture

# Fixed Infrastructure, Arbitrary Topologies

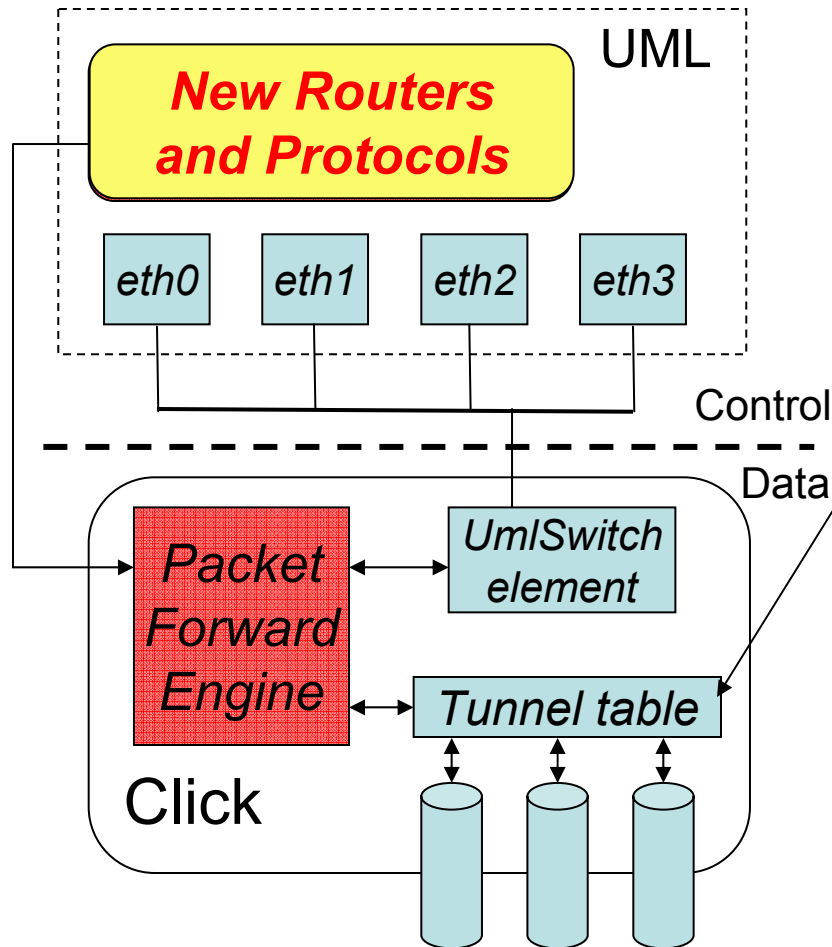


# Initial Deployment Platform: VINI



- XORP: control plane
- UML: environment
  - Virtual interfaces
- Click: data plane
  - Performance
    - Avoid UML overhead
    - Move to kernel, FPGA
  - Interfaces  $\Rightarrow$  tunnels

# First Step: Independence from IP



Forwarding cannot depend on IP

- **Solution:** Forwarding should depend on MAC addresses in UML

# Challenge #2: Substrate

- **Problem:** Brokering of physical infrastructure
  - **Discovery:** Discovering physical infrastructure
    - Autodiscovery of components and topology
    - Decision elements that configure components
  - **Provisioning:** Creating virtual networks
    - Requests to decision elements (initially out of band), which name virtual network components
  - **Creation:** Instantiating virtual networks

# Challenge: Embedding

- **Given:** virtual network and physical network
  - Topology, constraints, etc.
- **Problem:** find the appropriate mapping onto available physical resources (nodes and edges)
- Many possible formulations
  - Specific nodes mapping to certain physical nodes
  - Generic requirements: “three diverse paths from SF to LA with 100 MBps throughput”
  - Traffic awareness, dynamic remapping, etc.
  - On-the-fly creation of links in the substrate

# Parallel Deployment: Questions

- Guaranteeing global reachability
  - Do we need an end-to-end global reachability service?
- Proliferation of protocols and architectures
  - Is “low barrier to entry” a good thing for an architecture?
- Security
  - Should parallel deployment imply isolation?
  - If so, which way to implement it?

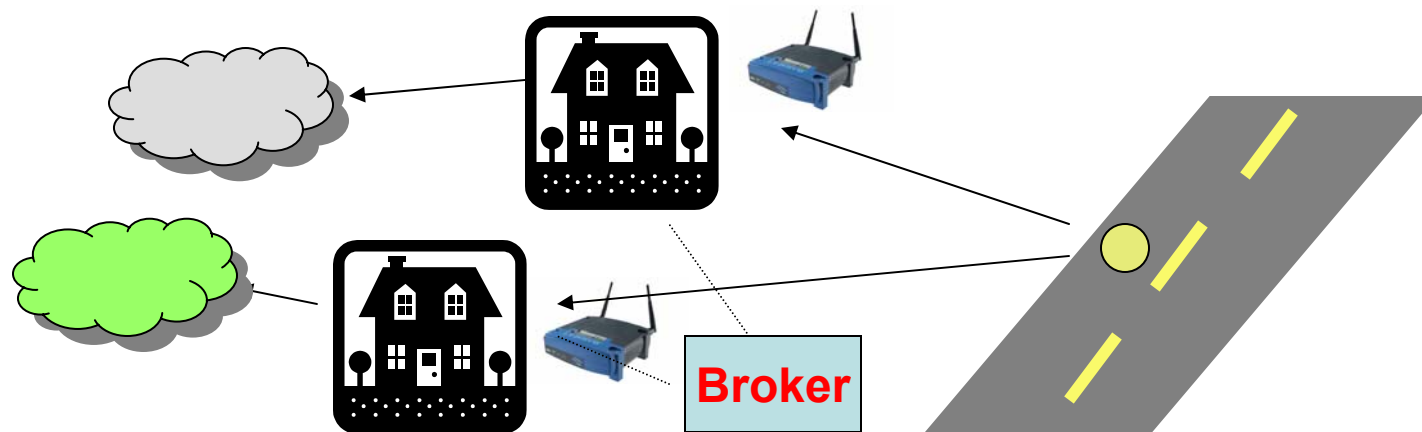
# Economic Refactoring

Whether this refactoring would occur is critical to many aspects of Cabo.

- Being a service provider: a great deal
  - Opportunity to add value by creating new services
- Infrastructure providers
  - Can this enterprise be profitable?
- Who will become infrastructure providers?

# Examples in Communications Networks

- **Packet Fabric:** share routers at exchange points
- **FON:** resells users' wireless Internet connectivity



- **Infrastructure providers:** Buy upstream connectivity, broker access through wireless
- **Nomads:** Users who connect to access points
- **Service provider:** FON as broker

# Summary

- Internet faces stronger demands from users
  - Not necessarily designed for those challenges
  - Difficult to deploy fundamentally new fixes
- Virtualization: help us move beyond paper design
  - Testbed
  - Architectural foundation
- Many challenges, both economic and technical
  - Embedding and resource allocation
  - Economic refactoring

<http://www.cc.gatech.edu/~feamster/papers/cabo-tr.pdf>