We Dream of GENI:
Exploring Radical Network Designs

Scott Shenker
UCB and ICSI

On behalf of many others

Terminology

radical = non-incremental

Designs derived from asking: "if we could redesign the Internet from scratch, what would we do?"

Talk Outline

- ***Why*** should we consider radical designs?
- ***What*** are some of these radical ideas?
- ***How*** can we test radical designs?

Three Obvious Statements

- We now live in a networked world
  - *Connecting* as important as *computing*
- The Internet is one of research's great triumphs
  - Original design a product of research, not industry
- The Internet is a victim of its own success
  - Has changed the standards by which it is judged…..

Changing Context and Expectations

- Internet architecture has been incredibly successful
  - Scaled many orders of magnitude in size and speed
  - Accommodated diversity of uses and technologies
  - Has changed the context in which it operates

- Led to requirements not met by original architecture
  - These requirements pose deep intellectual challenges
  - Not “how to patch”, but “how to design from scratch”

- *Understanding* requires rethinking basic paradigm
  - *Coping* may (not) require significant architectural changes

Environment: Trusted ⇒ Untrusted

- Requires a far more secure Internet
  - What do we mean by security?
  - What aspects are the network's responsibility?

- Major design challenges:
  - Resilience to large-scale external attacks (DDoS)
  - Resilience to compromised routers
  - Easy authentication of data
  - Forensics and auditing
  - Providing both accountability and privacy
  - …….

Users: Researchers ⇒ Customers

- Customers demand high availability
  - Service is almost never interrupted
- Internet was designed for strong recovery properties
  - Recovering from serious failures
- How can the Internet provide 5 9's of availability?
  - …and doing so in a cost-effective manner
  - Internet currently at 2-3 9's

Operators: Nonprofit ⇒ Commercial

- Operators must be able to manage their networks
  - Configuration
  - Troubleshooting
  - Middleboxes (proxies, firewalls, NATs, etc.)
  - Policy (routing, access control)

- What are the right abstractions for management?
  - What mechanisms best support them?

Usage: Host-oriented ⇒ Data-oriented

- Internet was designed around a host-oriented model
  - User tells client to contact another host (telnet, ftp)
- Current usage is mostly data-centric
  - User wants to access particular data or service
  - Does not care where that service is located
- Mismatch currently handled by *ad hoc* mechanisms
  - Akamai, P2P
- Right abstractions for a data-oriented Internet?

Connectivity: E2E IP ⇒ Intermittent X

- Architecture assumes end-to-end IP connectivity
- In some niche settings, each link is intermittent and end-to-end connectivity is rare
  - Space, underwater, developing economies
  - Led to call for “delay-tolerant networking” (DTN)
- More generally want to shield applications from networking details
  - Opportunistic and context-dependent communication
- What’s the right API to enable this generality?

New Grand Challenges

- Medicine:
  - All medical devices controlled over network
  - Security and reliability paramount
- Developing economies:
  - Little infrastructure or operational support
  - Must rely on self-organizing P2P-style designs
- Emergency response:
  - Rapid deployment and prioritized usage
  - Must operate under extreme conditions

Responding to These Requirements

- Could focus on incrementally-deployable changes
  - Might provide immediate, if partial, relief
  - Wouldn't know about long-term wisdom of changes

- Alternatively, we could think about the problem without constraints, with a "clean-slate"
  - Allows us to explore the conceptual underpinnings
  - Can later try to retrofit solutions onto the Internet

Clean-Slate

- Clean Slate is a means, not an end
  - No one expects direct adoption of radical ideas

- It is the insight that will have impact, by guiding the Internet's incremental evolution

  Clean-slate designs $\Rightarrow$ Insights $\Rightarrow$ Better Internet

- NSF's FIND program supports Clean-Slate research
  - Led by Dave Clark
  - See www.nets-find.net

Talk Outline

- ***Why*** should we consider radical designs?
  - The Internet is facing fundamentally new challenges
- ***What*** are some of these radical ideas?
- ***How*** can we test radical designs?

Improving Availability

- Routing algorithms with zero convergence time
  - Even right after failure, routing finds path to destination
  - Uses state in packet-header
- Packets sent along multiple paths
  - Traditional routing with “bits”
  - Diffusive routing with duplicate suppression on data path
- In both cases, only those clients needing high-availability are imposing burden on network

Making All Names Self-Certifying

- Self-certifying: derived from hash of public key
  - Use SCNs for: addresses, hosts, ASes, data, services,…
- Well-known technique, but embedding it in architecture would provide significant benefits
  - Authenticate data without PKI
  - Secure routing (without PKI or address registry)
  - Mitigate DDoS (with smart NICs)

Improved Name Resolution

- DNS currently resolves names by “look-up”
  - Hard to handle replication and locality (Akamai)
- Some proposals (TRIAD) resolve names by “routing”
  - Name servers keep name-based routing table
  - Resolution request is routed towards closest copy
  - Name servers also support caching and RSS
- Embeds basic CDN support into infrastructure
  - Application-independent
  - Scalable

Improving Management

- Centralize the control plane
  - Routers become “dumb” forwarding boxes
  - All control decisions are made by centralized controllers, which have global view of network
- Makes configuration and policy easy
  - No longer requires distributed algorithm to achieve
  - No need for complicated management abstractions
- Reliability achieved by standard replication

Living Without Congestion Control

- Congestion control is constant subject of study
  - But do we need it at all?
- Why not always send as fast as possible
  - Expect packet drops, use rateless encoding
  - Stop when data can be reconstructed
- Routers need no buffers, only need to provide some degree of fair dropping
- Automatically leverages multipath routing

Dynamic Links

- Canonical routing paradigm:
  - Find best paths over fixed set of links
  - Respond to failures, but changes in topology are rare
- New technologies can dynamically switch lambdas
  - Can establish new “links” very rapidly
  - Traffic engineering becomes a very different problem
  - Core routers become very simple optical devices
- Other ways “links” will become outmoded:
  - Wireless
  - Broadcast satellites

New API

- Applications should be shielded from details of communication
  - Should operate on names and application data units
  - Not on addresses and byte-streams
- Many have advocated a publish/subscribe interface
  - Application doesn’t know how data is served or obtained, merely states the name of the desired data
- Combines insights from DTN, Pub/Sub, Data-oriented, and many other efforts

Many More “Radical” Ideas

- These are just a few of the many ideas under discussion

- Motivated by “what is the right way to do this”, not “how can we patch the existing Internet”

Talk Outline

- ***Why*** should we consider radical designs?
  - The Internet is facing fundamentally new challenges
- ***What*** are some of these radical ideas?
  - The community has promising new designs
  - But they are all untested
- ***How*** can we test radical designs?

Current Networking Testbeds

- Production testbeds:
  - Can’t try radical network-level experiments
- Experimental testbeds:
  - No real users
  - Not much better than simulation
- Both kinds of testbeds:
  - Only one experiment at a time
  - Limited to sites directly connected to testbed
  - Hard to program

Leaves Us Unable to Evaluate Designs

- Conferences are littered with promising proposals
- But we can’t tell the good ideas from the bad
  - Because we never see them in operation at significant scale, with real traffic
- Architecture is no longer an experimental science
  - It has become *science fiction*
- Given challenges we face, this must be overcome

The Testbed We'd Wish For

- Usable by many experiments simultaneously
- Easily programmable
- Can experiment on any level (optical to apps)
- Users can "opt-in" even from remote locations
- Reasonably large scale

GENI Will Grant Our Wishes

- GENI: Global Environment for Network Innovations
  - Project being proposed to NSF
- If approved, would be funded by NSF's Major Research Equipment and Facilities Construction (MREFC) account
  - MREFC is used to fund large experimental facilities
  - Telescopes, research vessels, etc.
- First MREFC initiated by computer science

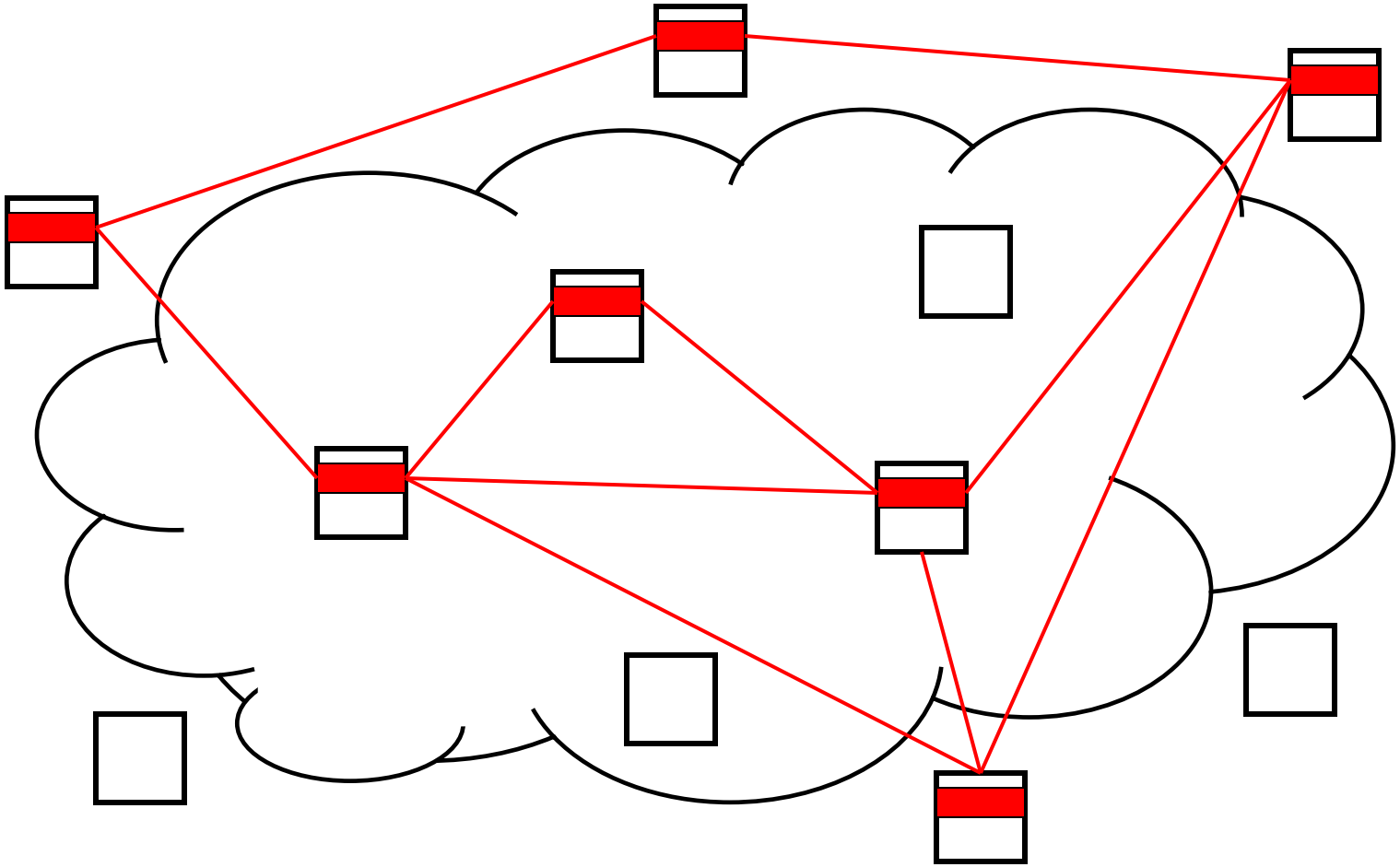
GENI Design Principles

- An generalization of the PlanetLab approach…
- GENI is comprised of network resources
  - Links, nodes, subnets,…
- Resources are *virtualizable* and *programmable*
  - Can be partitioned among many researchers
  - Can implement radical new designs
- Researchers can program GENI at any level of abstraction
  - Optical, IP, application,….

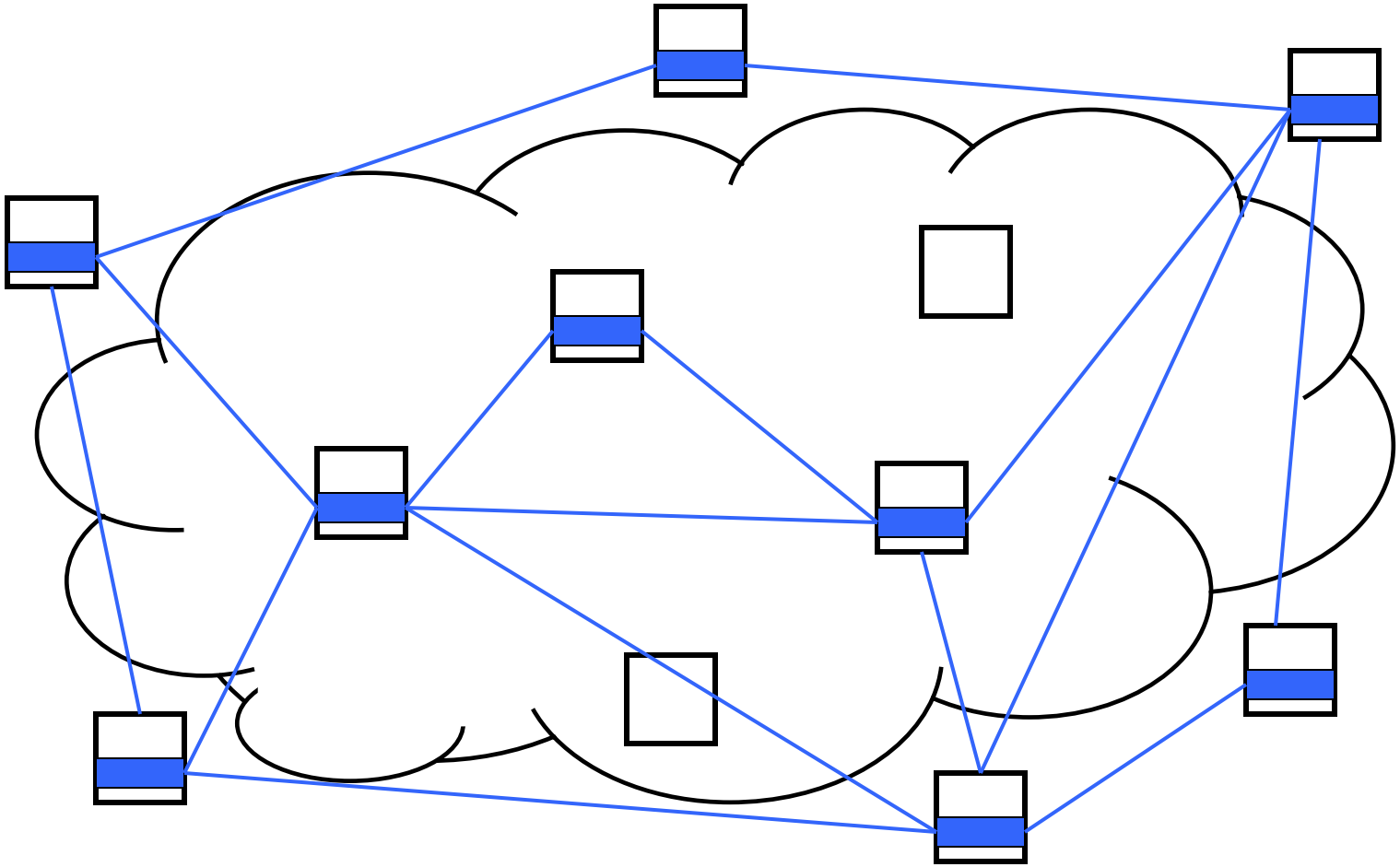
GENI Design Principles (cont'd)

- Wide variety of networking technologies
  - Optical, wireless, sensors, phones,…
- Large-scale (~25 PoPs)
- Users can access GENI through overlay

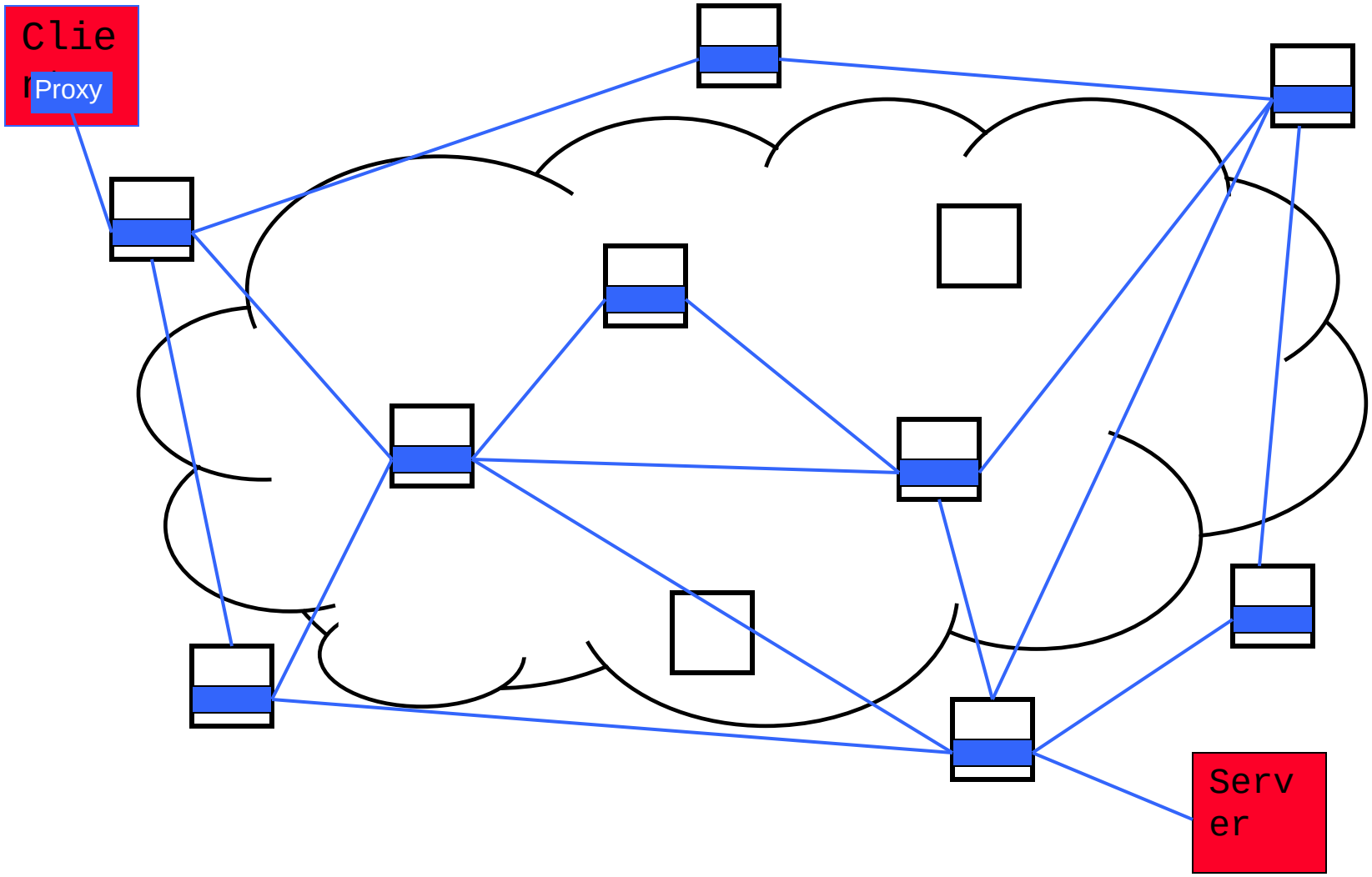
Each Researcher Gets a “Slice”



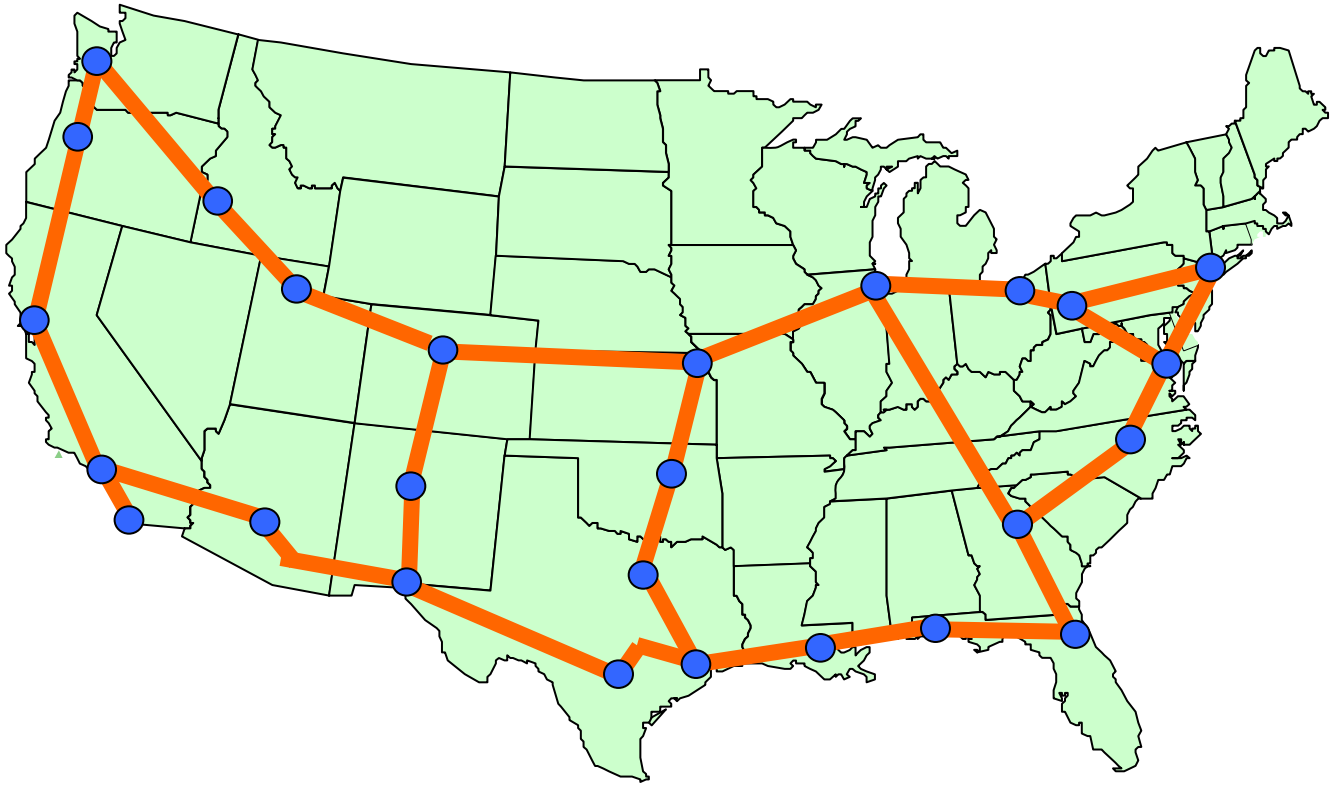
And They Don’t Interfere



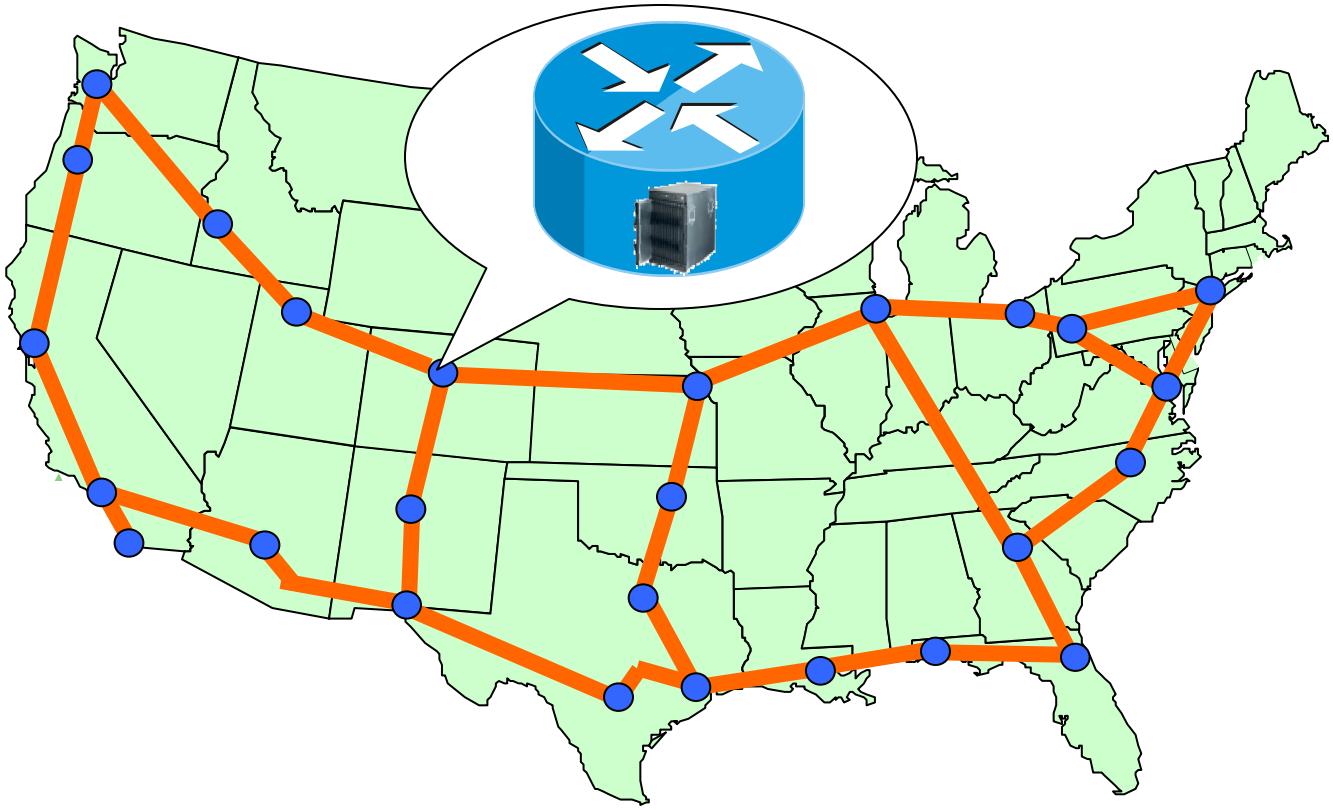
User Opt-in



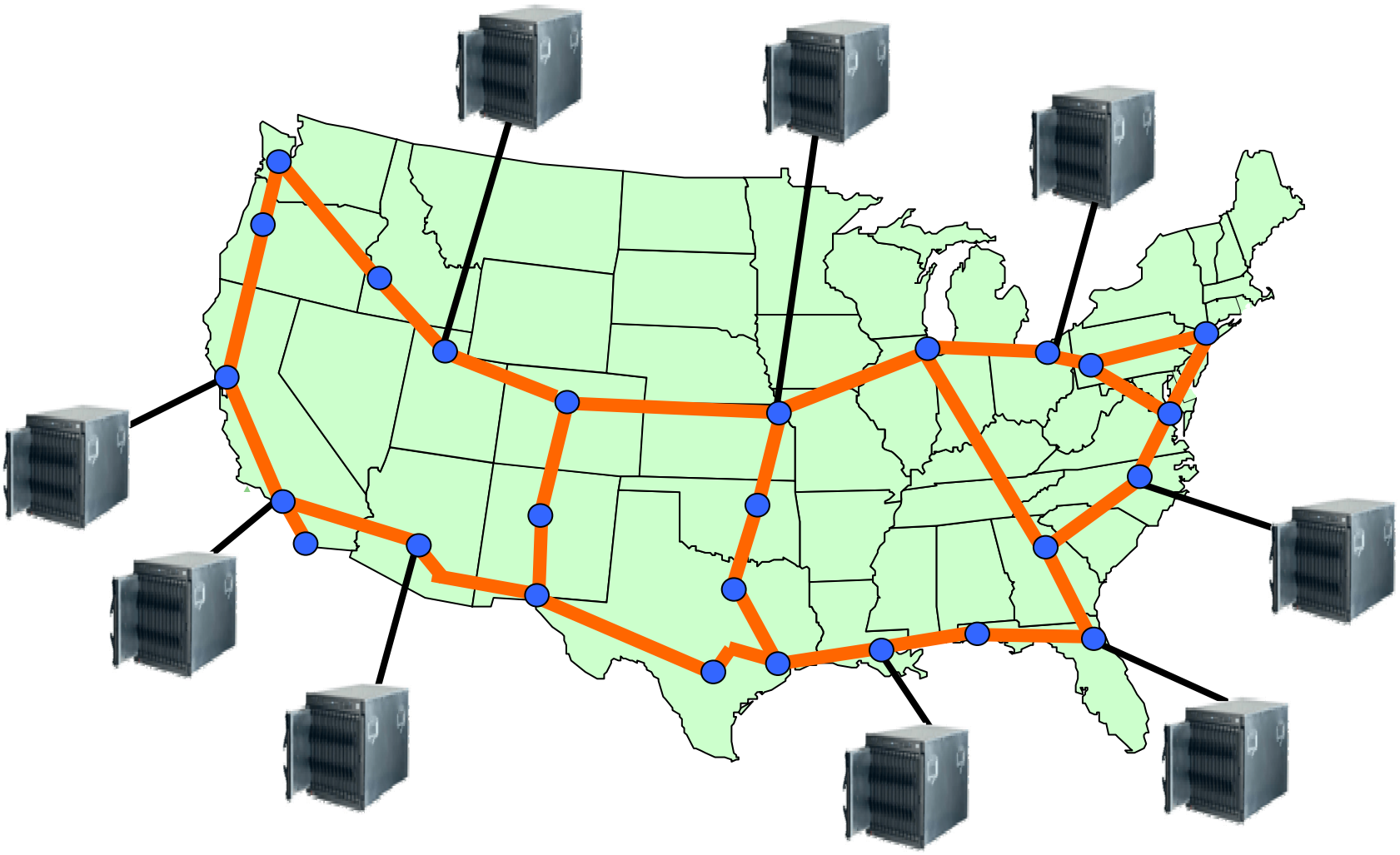
National Fiber Facility



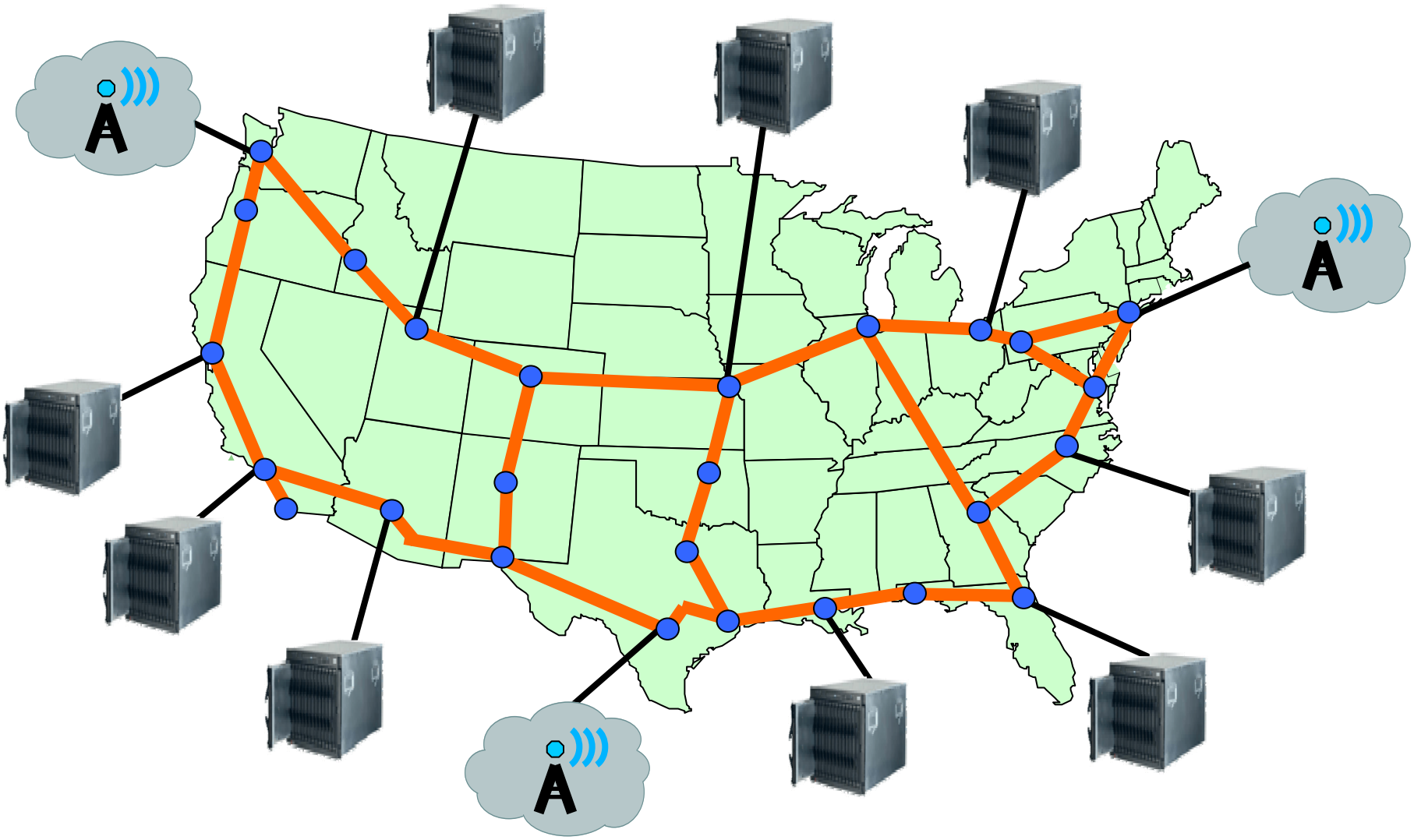
+ Programmable Routers



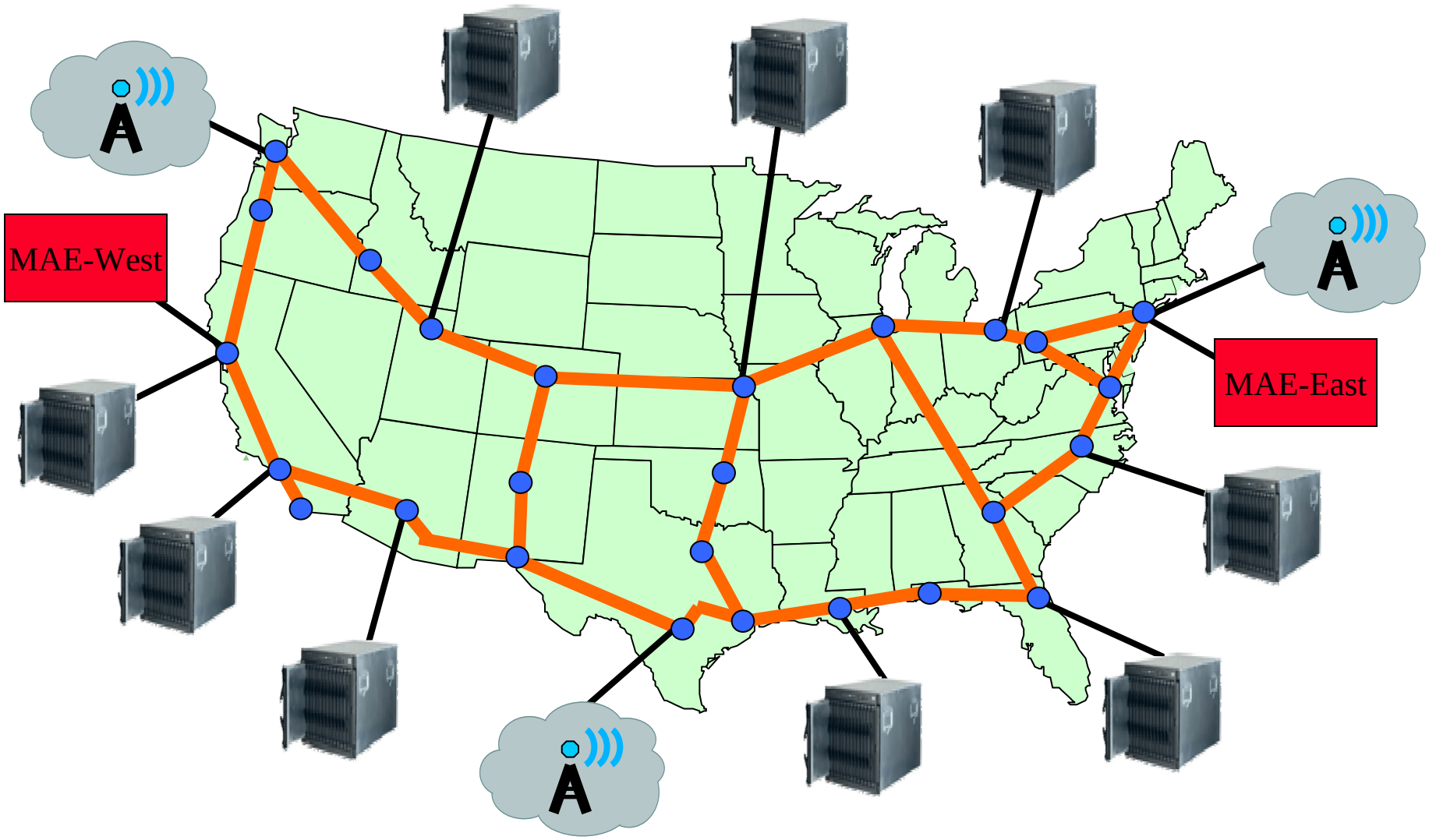
+ Clusters at Edge Sites



+ Wireless Subnets



+ ISP Peers



GENI Will Enable Us To…

- Experiment at scale
- 1000s of simultaneous experiments
- Long-running services (operational experience)
- Integrate our designs across layers

Not Just for Networking!

- GENI originally motivated by networking agenda
- But can support a much wider research agenda:
  - Distributed systems
  - New applications
  - User studies
  - …..
- Today's talk was not about GENI's breadth, but about how much networking needs GENI

GENI Status

- GENI still in planning stage
  - Public workshop to be held in September
  - Call for whitepapers out by end of June
- Relevant bodies:
  - Interim planning group (was led by Larry Peterson)
  - GENI Science Council (chaired by Ellen Zegura and SS)
  - GENI Project Office (BBN, led by Chip Elliot)
- See www.geni.net (soon to be updated)

Summing Up

- We have a technical vision
  - Practically important and intellectually deep problems posed by new networking challenges
- We have funding for this vision
- We have prospects for an experimental facility
- But this is not enough!

Need Community Commitment

- Architecture is not simple sum of 300 papers
  - Product of broad synthesis and collaboration
  - Not your traditional academic behavior
- Community must be committed to working together to create a few shared visions of the future
  - Design
  - Build
  - Operate
- The FIND program is building that commitment

“Perfect Storm” Brewing in Networking

- Commitment to a “grand agenda”
  - Technical ambition: rethinking the Internet
  - Community commitment to work together
- Prospects for experimental facility
  - Learn by building and using, not just paper designs
  - Return architecture to its roots as an experimental science
- Conclusion: very exciting time in networking!