GENI Backbone: Design Overview

GENI Backbone Working Group

Dan Blumenthal, Dean Casey, Patrick Crowley, Farnam Jahanian, T.V. Lakshman, Bryan Lyles, Nick McKeown, Sampath Rangarajan, Jennifer Rexford, Jon Turner, and Hui Zhang

Outline

- Backbone requirements
  - Principles
  - Experimenter's view
- Phased deployment
  - General-purpose processors to NPs and FPGAs
  - Conventional optics to programmable components
- Software modules
  - Component managers, libraries, etc.
- Example experiments
  - End-system multicast, routing control platform, valiant load balancing, and flow switching
- Recent issues discussed in the backbone group

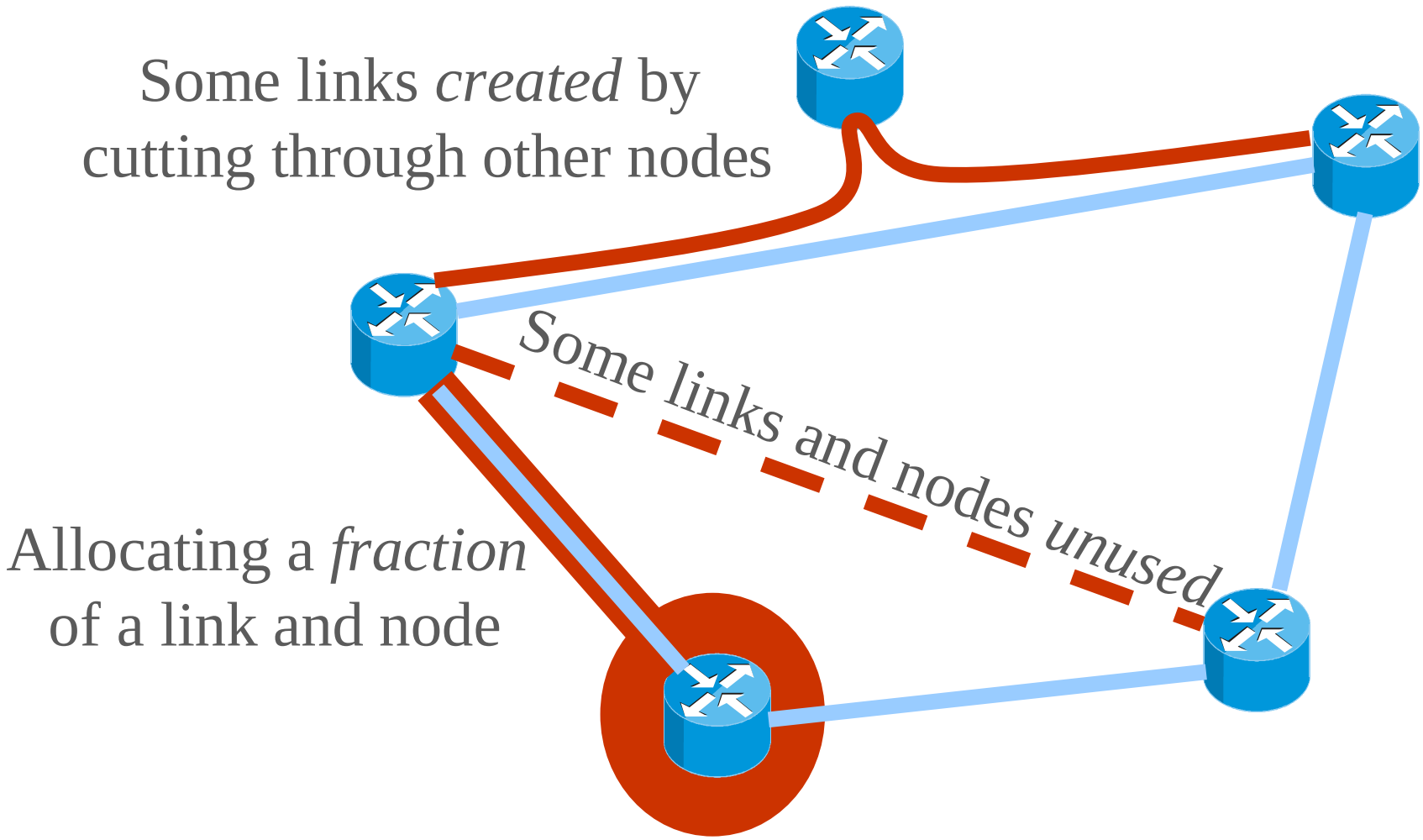
GENI Backbone Requirements

- Programmability
  - Flexible routing, forwarding, addressing, circuit set-up, …
- Isolation
  - Dedicated bandwidth, circuits, CPU, memory, disk
- Realism
  - User traffic, upstream connections, propagation delays, equipment failure modes
- Control
  - Inject failures, create circuits, exchange routing messages
- Performance
  - High-speed packet forwarding and low delays
- Security
  - Preventing attacks on the Internet, and on GENI itself

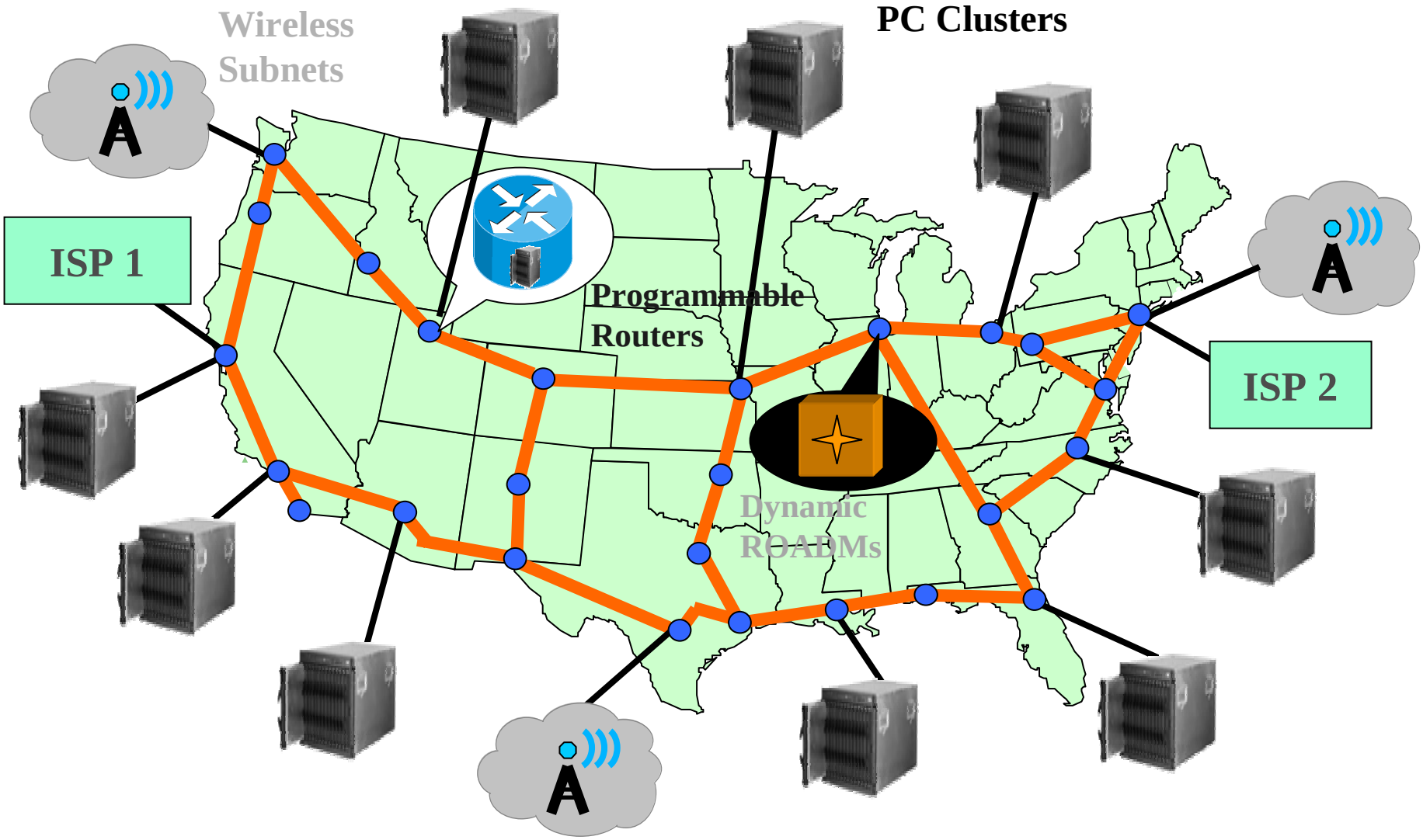
A Researcher's View of GENI Backbone

- Virtual network topology
  - Nodes and links in a particular topology
  - Resources and capabilities per node/link
  - Embedded in the GENI backbone
- Virtual router and virtual switch
  - Abstraction of a router and switch per node
  - To evaluate new architectures (routing, switching, addressing, framing, grooming, layering, …)
- GENI backbone capabilities evolve over time
  - To realize abstraction at finer detail
  - To scale to large # of experiments

Creating a Virtual Topology



GENI Backbone

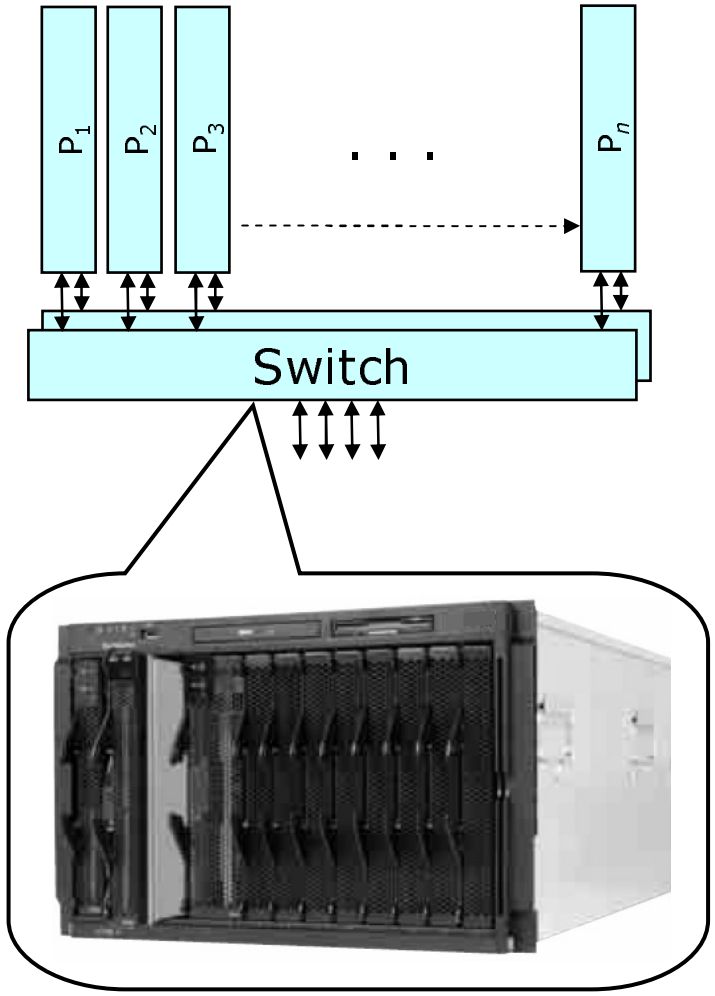


GENI Backbone: Phased Deployment

- Phase 0: general purpose processors
  - General purpose processors connected to switch
  - Virtualization platform for multiple virtual routers
- Phase 1: ATCA-based programmable router
  - Network processors and FPGAs, and line cards
  - Faster packet processing and line-card cut through
- Phase 2: reconfigurable optics
  - Cross-connect and off-the-shelf framer/groomer
  - True circuits and bypass of programmable router
- Phase 3: programmable optics
  - Dynamic optical switch with programmable framer
  - Experimental flexibility for framing, grooming, set-up, …

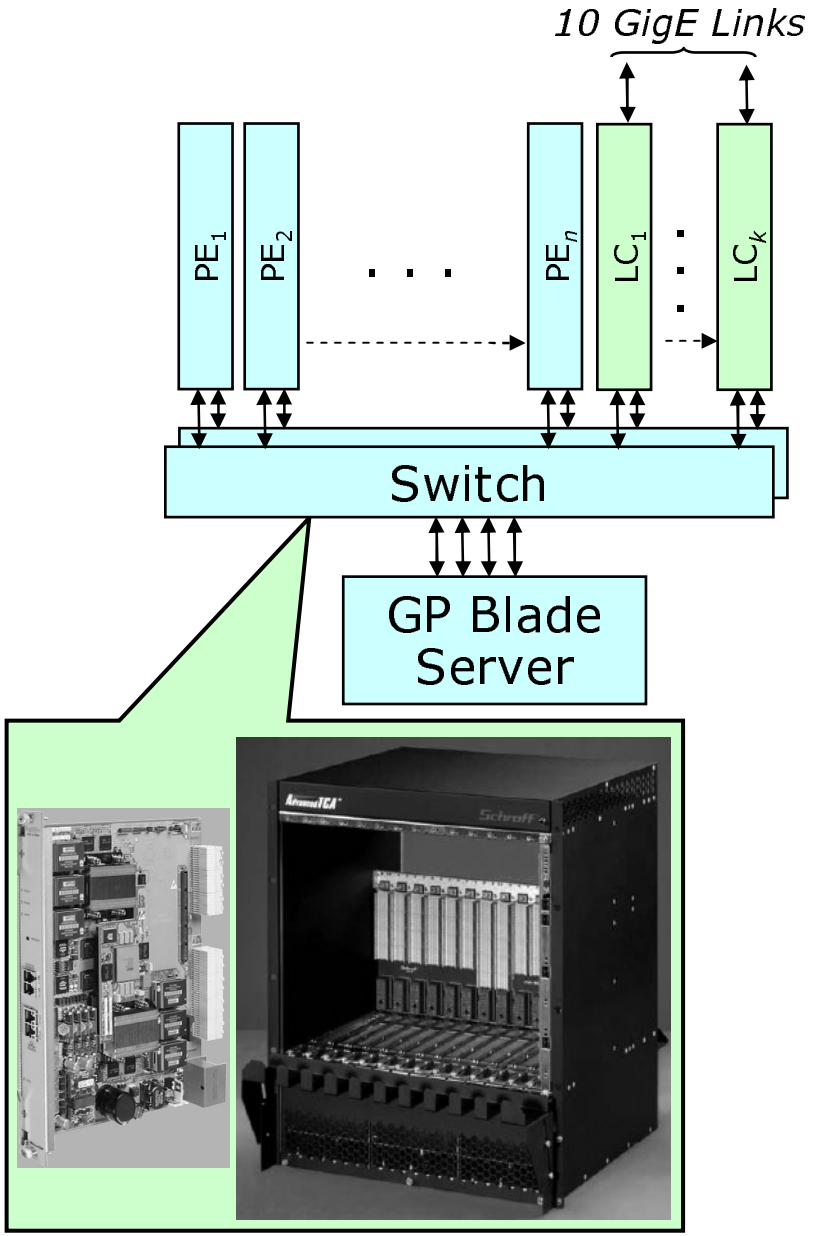
Phase 0: General Purpose Processors

- Node with collection of assignable resources
- Embed virtual routers
  - A virtual machine
  - An entire machine
  - Multiple machines
- Initial VINI deployment
  - In NLR and Abilene
- Ultimately not scalable
  - All packets go through the CPU
  - Slow software packet processing



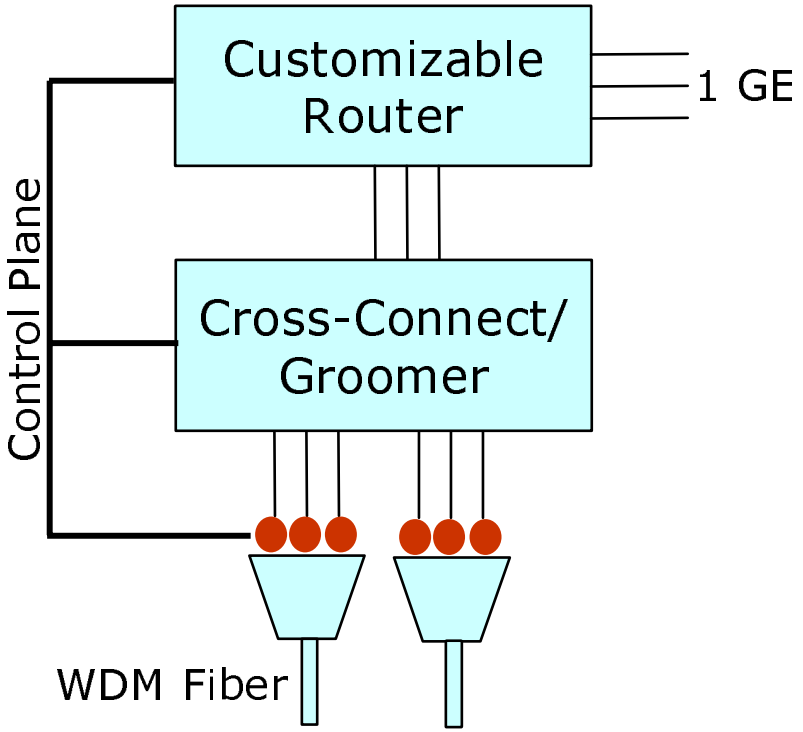
Phase 1: High-Performance Components

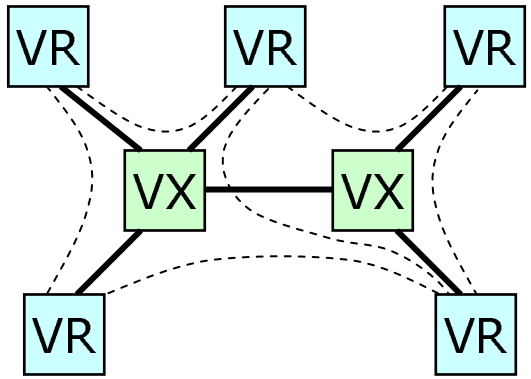
- ATCA chassis and blades
- Network Processor (NP) and FPGA blades
  - Offload packet handling
  - Control plane on GPs
- Line-card cut-through
  - For a slice not running at this site
  - Packets go from LC to LC



Phase 2: Reconfigurable Optics

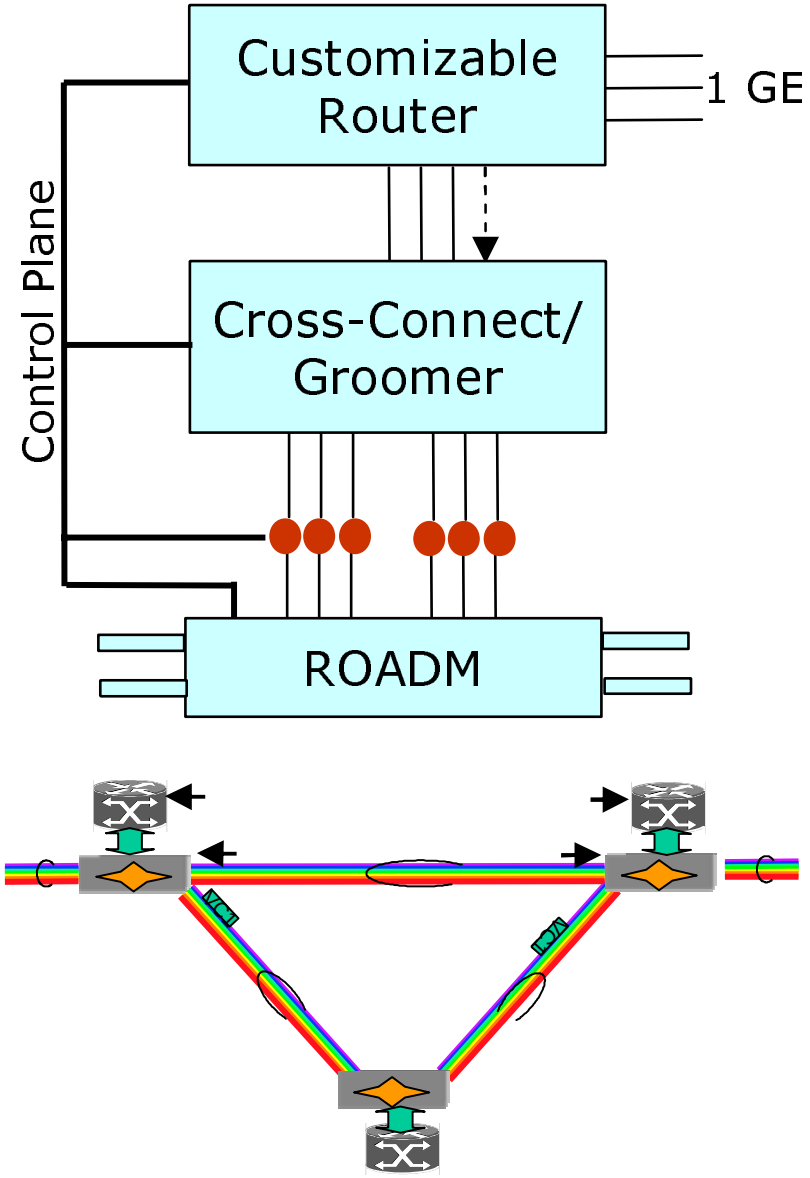
- Circuits between backbone routers
  - Cut-through traffic completely bypasses programmable router
- Conventional framing
  - E.g., SONET
- GENI controls circuit set-up and teardown
  - As part of creating and deleting a slice
  - Limited flexibility



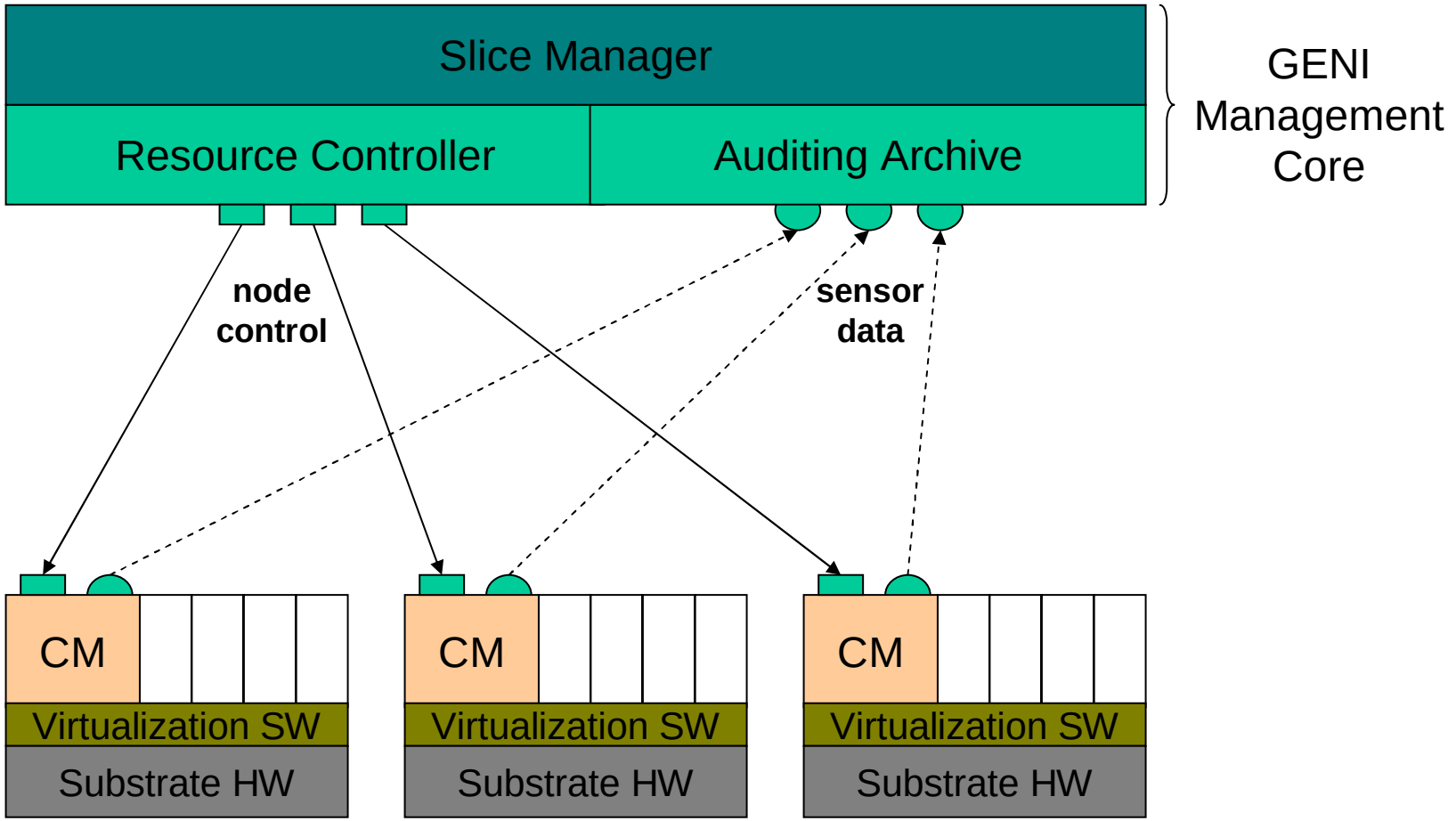


Phase 3: Programmable Optics

- Dynamic optical switch
  - Programmable groomer and framer
  - Reconfigurable add/drop multiplexers
- Experimental flexibility
  - Maleable bandwidth
  - Arbitrary framing implemented on FPGAs
  - Dynamic set-up and teardown of circuits



Software: GENI Management Core



GENI Backbone Software

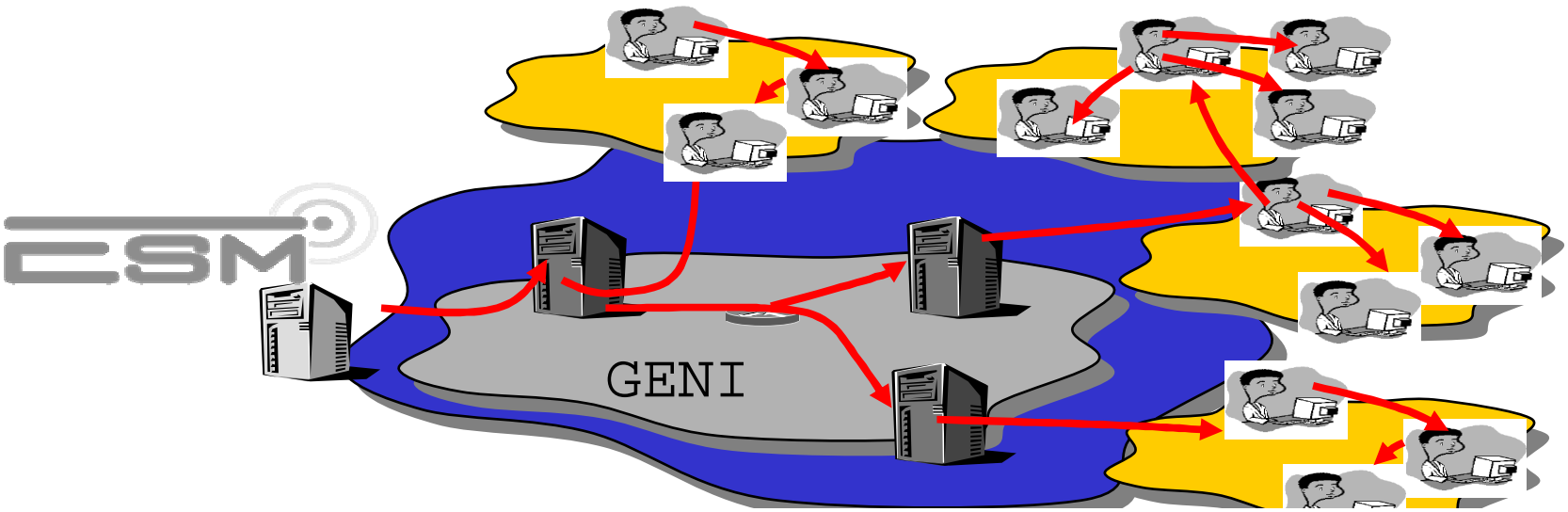
- Component manager and virtualization layer
  - Abstraction of virtual router and virtual switch
  - Setting the scheduling parameters
- Multiplexers for resources hard to share
  - BGP sessions with the outside world
  - Interface to element-management systems
- Exchanging traffic with the outside world
  - Routing and forwarding software
  - VPN servers and NATs at Internet boundary
- Libraries to support experimentation
  - APIs for access to NP-based packet forwarding
  - Specifying, controlling, and measuring experiments
  - Auditing and accounting to detect misbehavior

Feasibility

- Industrial trends and standards
  - Advanced Telecom Computing Architecture (ATCA)
  - Network processors and FPGAs
  - SONET cross connects and ROADMs
- Open-source networking software
  - Routing protocols, packet forwarding, network address translation, diverting traffic to an overlay
- Existing infrastructure
  - PlanetLab nodes, software, and experiences
  - National Lambda Rail and Abilene backbones
  - Ongoing work on VINI and programmable router

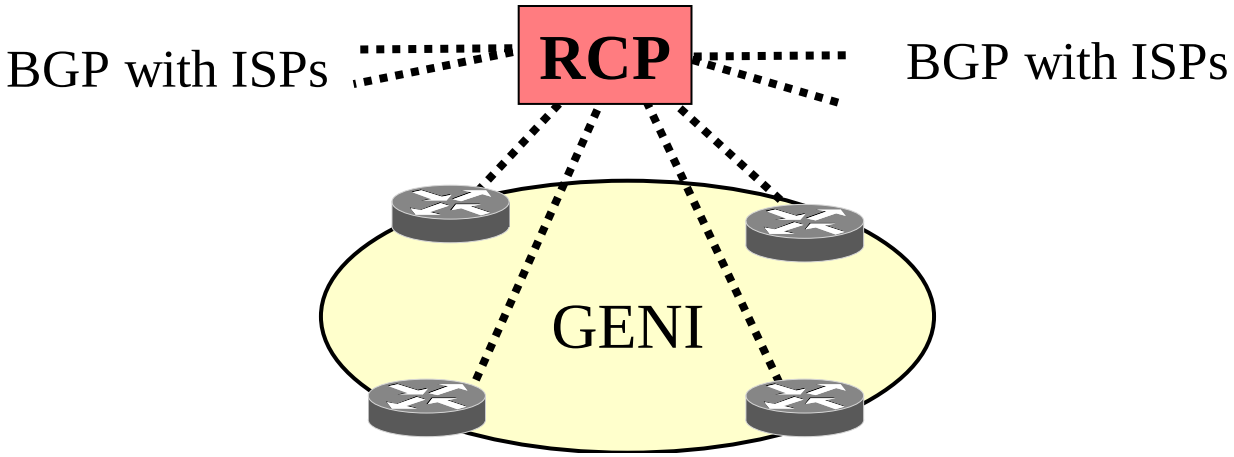
Example Experiments: End-System Multicast

- End-System Multicast
  - On-demand, live streaming of audio/video to many clients
  - Intermediate nodes forming a multicast tree
- Ways GENI could support ESM research
  - Communication with Internet clients and servers
  - Backbone nodes participating in the multicast tree
  - New network architectures running under ESM
    - Native multicast support and QoS guarantees for live service
    - Burst transfer, push, and network-storage for on-demand service



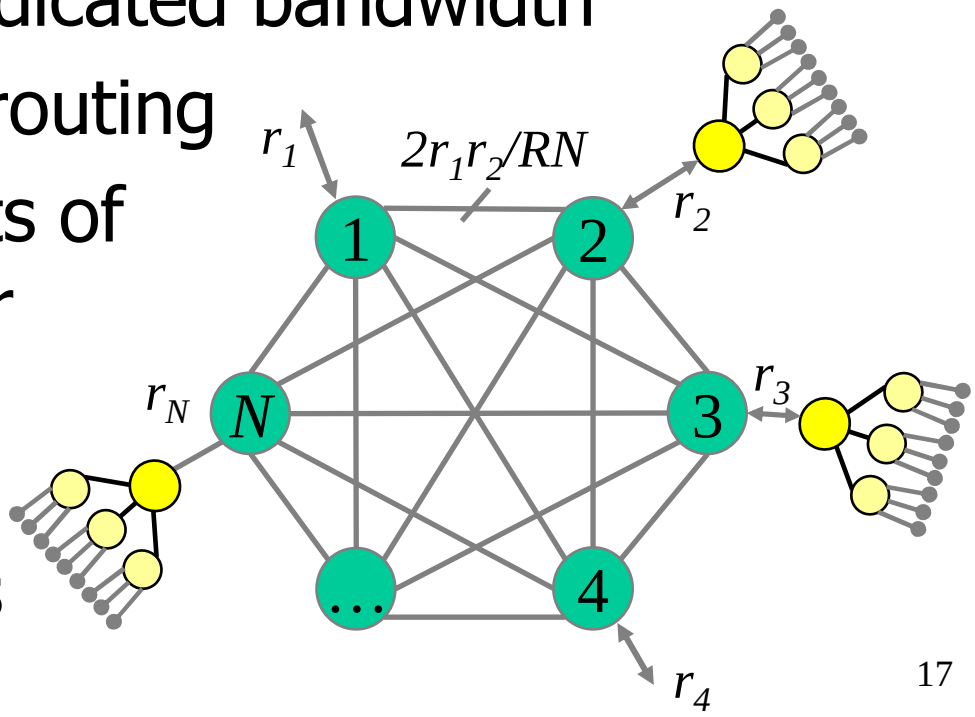
Example Experiments: Routing Control Platform

- Routing Control Platform (RCP)
  - Refactoring of control and management planes
  - Computes forwarding tables in separate servers
- Ways GENI can support RCP research
  - Providing direct control over the data plane
  - BGP sessions with the commercial Internet
  - Controlled experiments with node/link failures



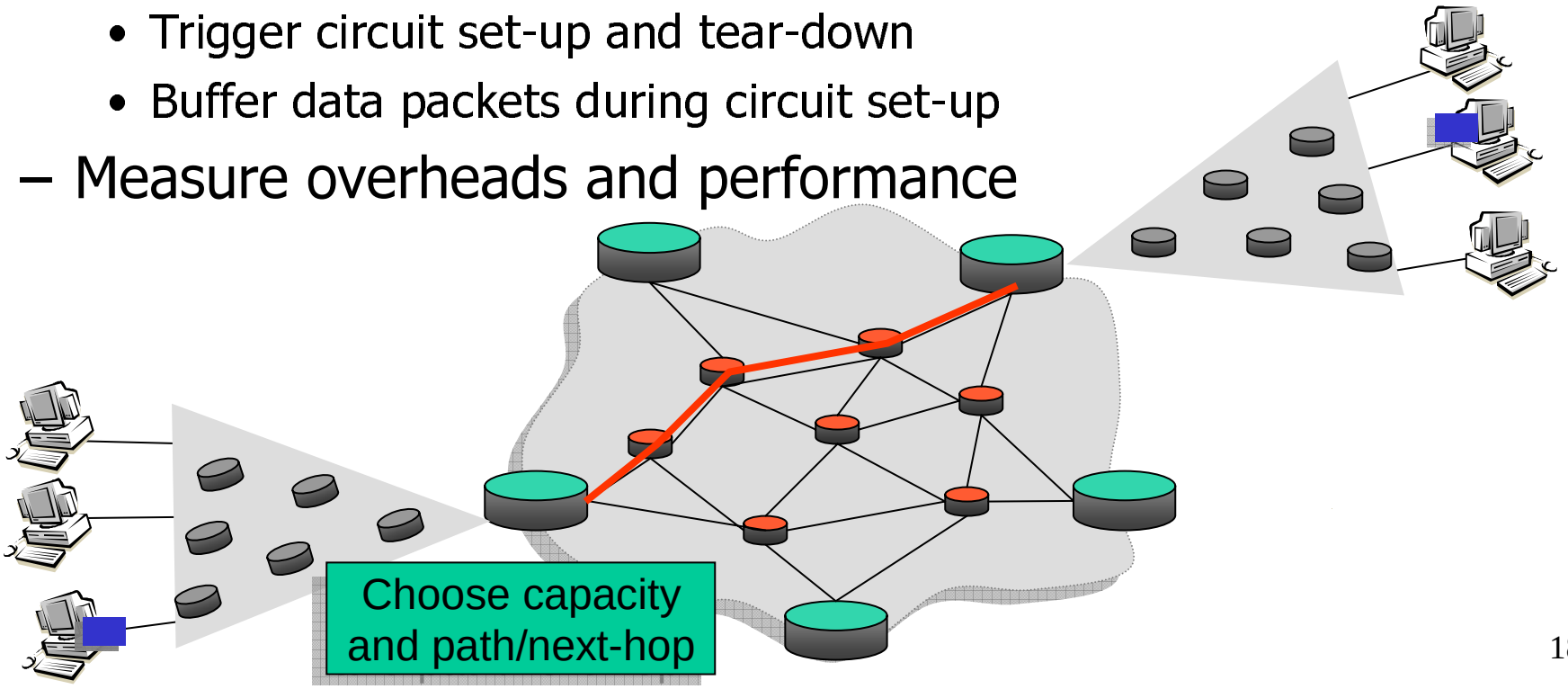
Example Experiments: Valiant Load Balancing

- Valiant Load Balancing
  - Fully mesh of circuits between routers
  - Direct traffic through intermediate node
- Ways GENI can support VLB
  - Virtual circuits with dedicated bandwidth
  - Experimentation with routing
  - Measurement of effects of higher delay vs. higher throughput on users
  - Explore impact on buffer sizing in routers



Example Experiments: TCP Switching

- TCP switching
  - TCP SYN packet triggers circuit set-up
  - Effective traffic management and quality of services
- Ways GENI can support TCP flow switching
  - Programmable routers act as edge routers
    - Trigger circuit set-up and tear-down
    - Buffer data packets during circuit set-up
  - Measure overheads and performance



Recent Issues Discussed in Backbone Group

- GENI topology design with 20-30 sites
  - Bounded delay, close to physical distance
  - Path diversity needed by many experiments
  - Matching fiber paths and interconnection points
  - Minimizing the backhaul cost of tail circuits
- Edge functionality
  - Need for packet scheduling at multiplexing points
  - To provide strong isolation between slices
  - Creation of use cases with the wireless group
  - E.g., CBR circuit between two access points

Recent Issues Discussed in Backbone Group

- Common software across working groups
  - Software for the general-purpose CPUs
  - Choice of operating system (e.g., Linux, real-time)
  - Virtualization layer for slivers and virtual interfaces
  - Scheduling/shaping for access to CPU, link, …
- Packet headers
  - Media header: per-hop header to traverse link
  - Sliver/v-interface headers: to de/mux to the slice
  - Experiment header: local to the experiment
  - Some headers may be combined on any given hop, and header formats may differ per hop

Recent Issues Discussed in Backbone Group

- Signaling vs. configuration to create slice
  - Need for packet de/multiplexing at each hop
  - Some sort of "label swapping" based on sliver
  - Set up these tables by top-down configuration?
  - Or, use some sort of signaling protocol?
  - Challenges of signaling a *graph*, rather than path
- Circuit abstraction at the edge?
  - Should edge sites be able to set-up circuits?
  - Or, can circuits only start at the backbone node?
  - Many experiments want a backbone cut-through?
  - Campuses increasingly have optical capabilities

Existing GENI Design Documents

- GDD-06-09
  - "A Proposed Architecture for the GENI Backbone Platform"
- GDD-06-25
  - "Backbone Software Architecture"
- GDD-06-26
  - "GENI Backbone Network Node Architecture: Requirements and Architecture"
- GDD-06-27
  - "GENI Topology Design"
- GDD-06-31
  - "In VINI Veritas: Realistic and Controlled Network Experimentation"

Planned GENI Design Documents

- Split software into several documents
  - Hardware-specific software for programmable router
  - Component manager and component aggregate
  - Run-time software for experiments
- Backbone facility design
  - Evolve "GENI Topology Design" into a more detailed set of requirements for the facility
  - Include requirements for tail circuits and how the optics connects to the underlying fiber plant

Conclusions

- Supporting experiments across the layers
  - VPN-like support for edge experiments
  - New control plane with packet forwarding offload
  - New addressing and forwarding schemes
  - Dynamic control of the optical components
- Balancing risk and flexibility
  - Phased deployment
  - Standards (e.g., ATCA, SONET)
  - Open-source software
  - Ongoing work on VINI and programmable router