Design Issues for the GENI Backbone Platform

Jon Turner
jon.turner@wustl.edu
http://www.arl.wustl.edu/arl

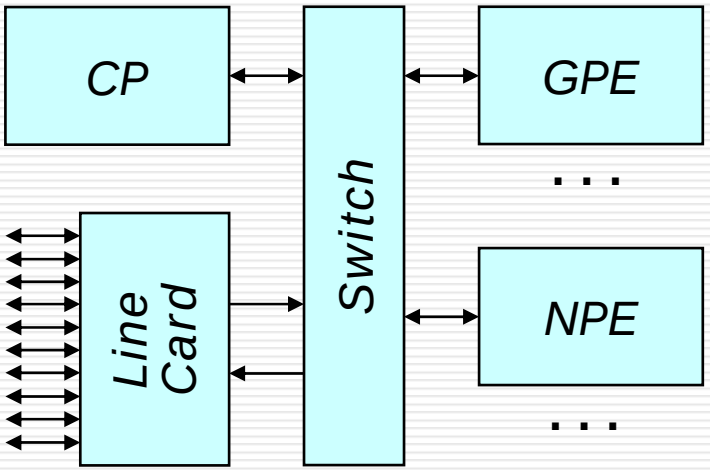
Overview

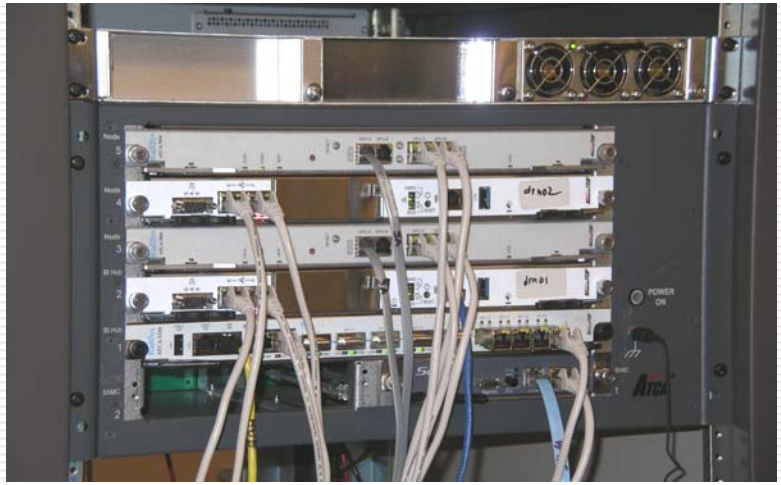
- High performance platform for overlay hosting services
  - » current focus on PlanetLab
  - » hardware is fully assembled and operational
  - » developed method to allow slices to share NP blades
  - » order-of-magnitude performance gains for two applications
- Proposed design for GENI backbone platform
  - » overall architecture and realization using ATCA
  - » potential for scaling it up
  - » how it might relate to flexible optical layer
  - » missing pieces
  - » things to be done
- Open Network Lab and GENI
- Thoughts about requirements
  - » balancing desires/needs with what's feasible/sensible

High Performance Overlay Hosting

- An *overlay hosting service* is a distributed infrastructure for hosting overlay networks
  - »PlanetLab is prototypical model
- To be useful as large-scale deployment vehicle, must
  - »handle Internet-scale traffic volumes
  - »deliver consistently higher levels of performance
- Integrated high performance platforms needed
  - »throughput of PlanetLab nodes tops out in 50-100 Kp/s range
    - versus 3.7 Mp/s for comparable Network Processor (NP)
  - »latency for PlanetLab apps ranges from 1-500 ms
    - versus 200 μs or less for NP
- Add provisioned links for managed QoS
  - »services for rapidly provisioning SONET, MPLS for backbone
  - »extend to end users, with provisioned VLANs on metro-Ethernet

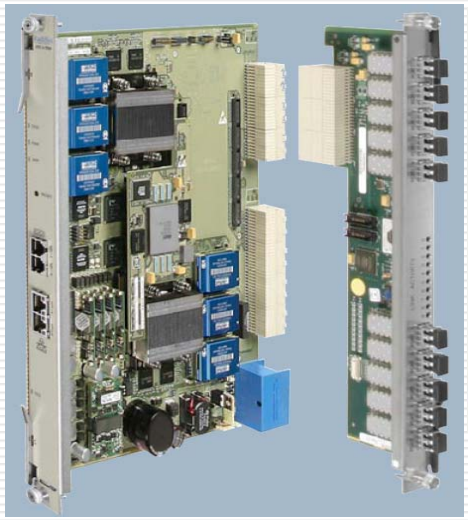
Supercharged Planetlab Platform (SPP)

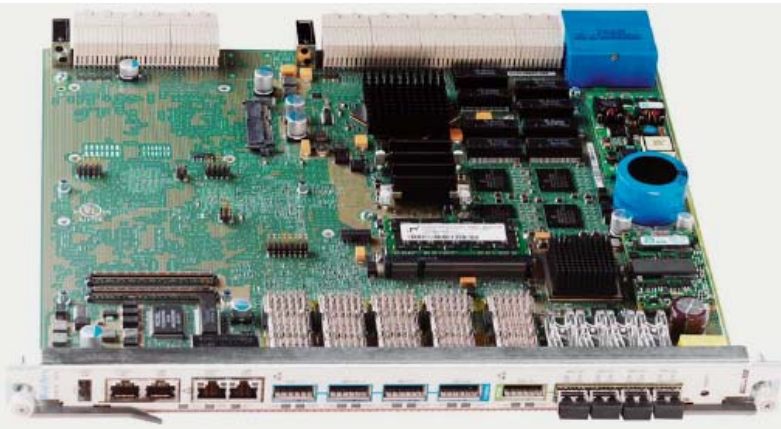




- Objectives
  - »enable order-of-magnitude improvements in throughput and latency for substantial subset of PlanetLab applications
- System components
  - »Line Card (LC) with 10x1GbE physical interfaces – (or 1x10GbE)
  - »Control Processor (CP) for system management
  - »Processing Engines for hosting slices
    - conventional server blade (GPE)
    - network processor blade (NPE)
- Switch blade – 10 GbE fabric plus 1 GbE control

Board-Level Subsystems

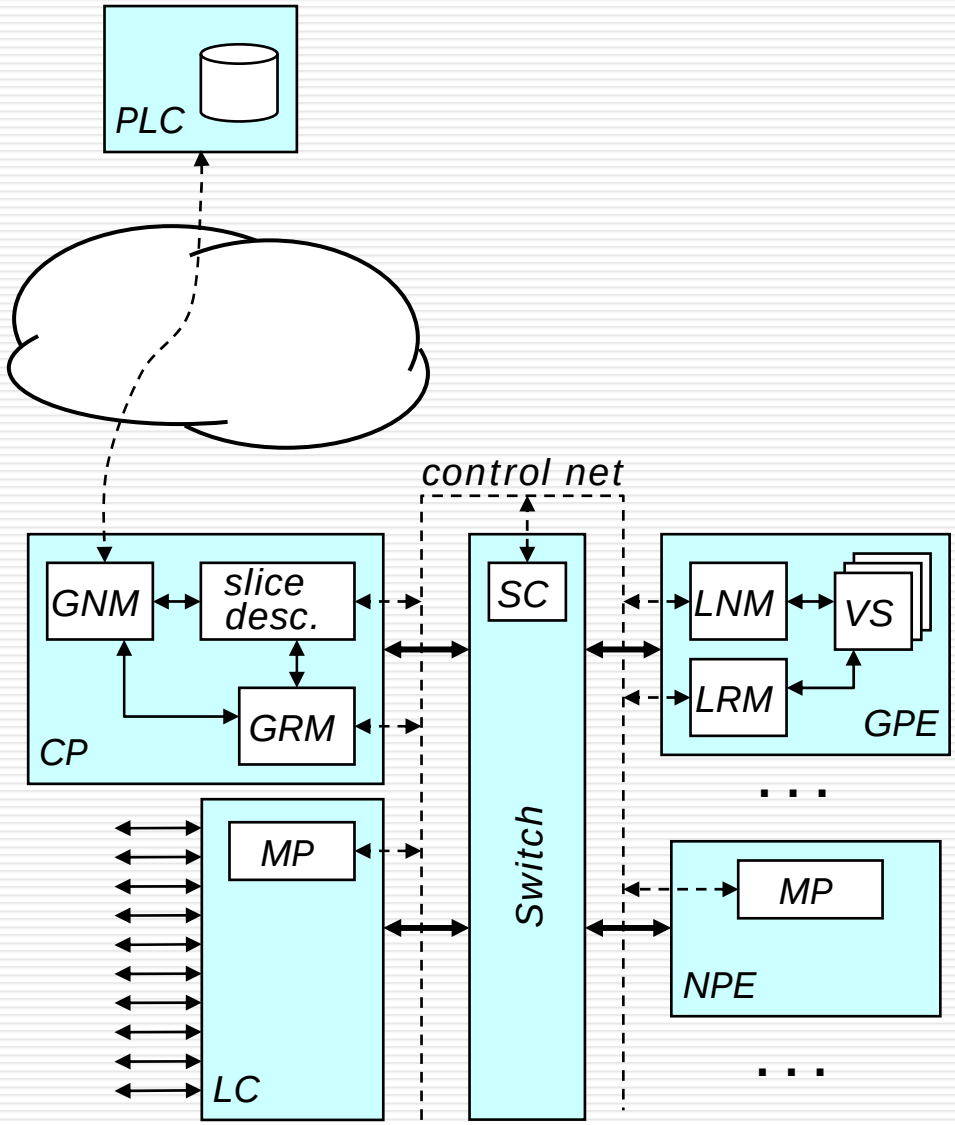




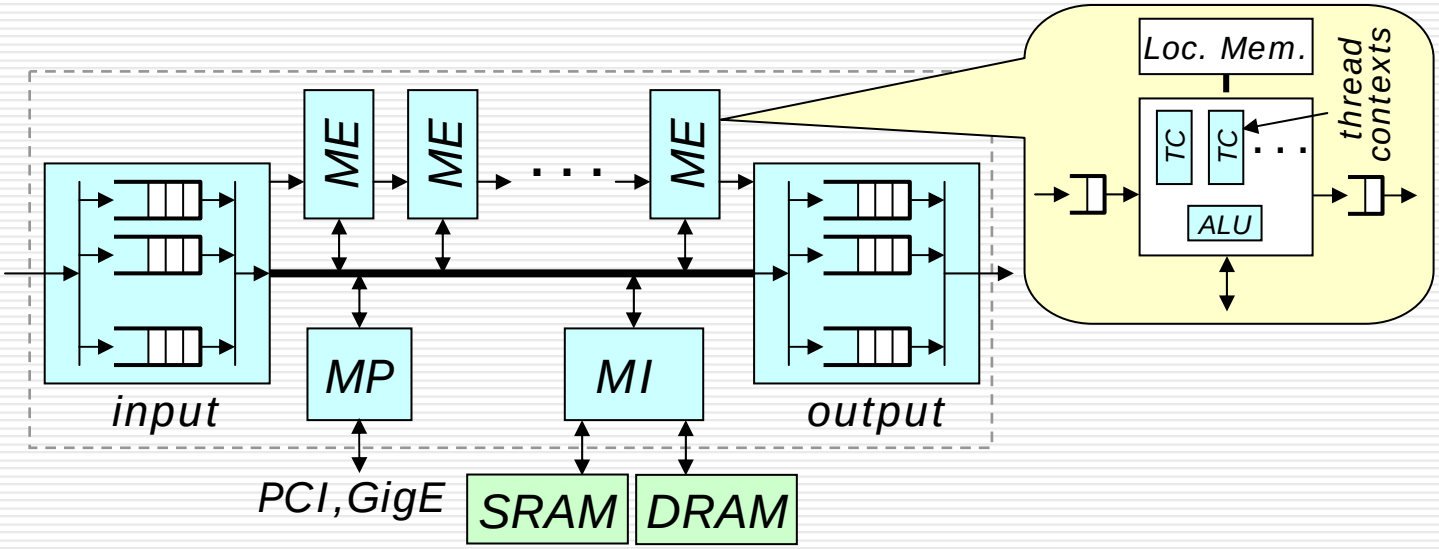
- Intel server blades
  - for CP and GPE
  - dual Xeons (2 GHz)
  - 4x1GbE
  - on-board disk
  - Advanced Mezzanine Card slot
- Radisys NP blades
  - for LC and NPE
  - dual IXP 2850 NPs
    - 3xRDRAM
    - 4xSRAM
    - shared TCAM
  - 2x10GbE to backplane
  - 10x1GbE external IO (or 1x10GbE)
- Radisys switch blade
  - up to 16 slot chassis
  - 10 GbE fabric switch
  - 1 GbE control switch
  - full VLAN support
- Scaling up
  - 5x10 GbE to front
  - 2 more to back

Overview of Control

- Illusion of single Plab node
- *Global Node Manager* (GNM)
  - » pools PLC for slice info
  - » stores copy of local slices
  - » selects GPE for new slice
- *Local Node Managers* (LNM)
  - » check for new slices on CP
  - » setup vServers (VS)
- Slices request NPE resources from *Local Resource Mgr* (LRM)
  - » assigned a *fast path slice*
  - » logical external interfaces defined by UDP port numbers
- Line Card (LC) manages mux/ demux using port numbers
  - » RMs coordinate “server ports”
  - » NAT for “client ports”



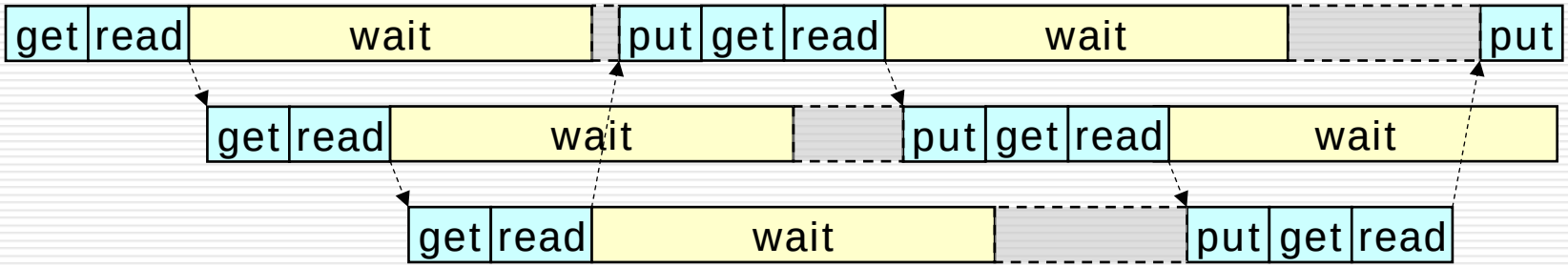
IXP 2850 Overview



- 16 multi-threaded Microengines (ME)
  - »8 thread contexts, with rapid switching between contexts
  - »fast nearest-neighbor connections for pipelined apps
- 3 SDRAM and 4 SRAM channels (optional TCAM)
- Not designed to be shared – no hardware protection
- Management Processor (MP) for control
  - »runs Linux and has “out-of-band” network interface
  - »start/stop/reprogram MEs, read/write all memory

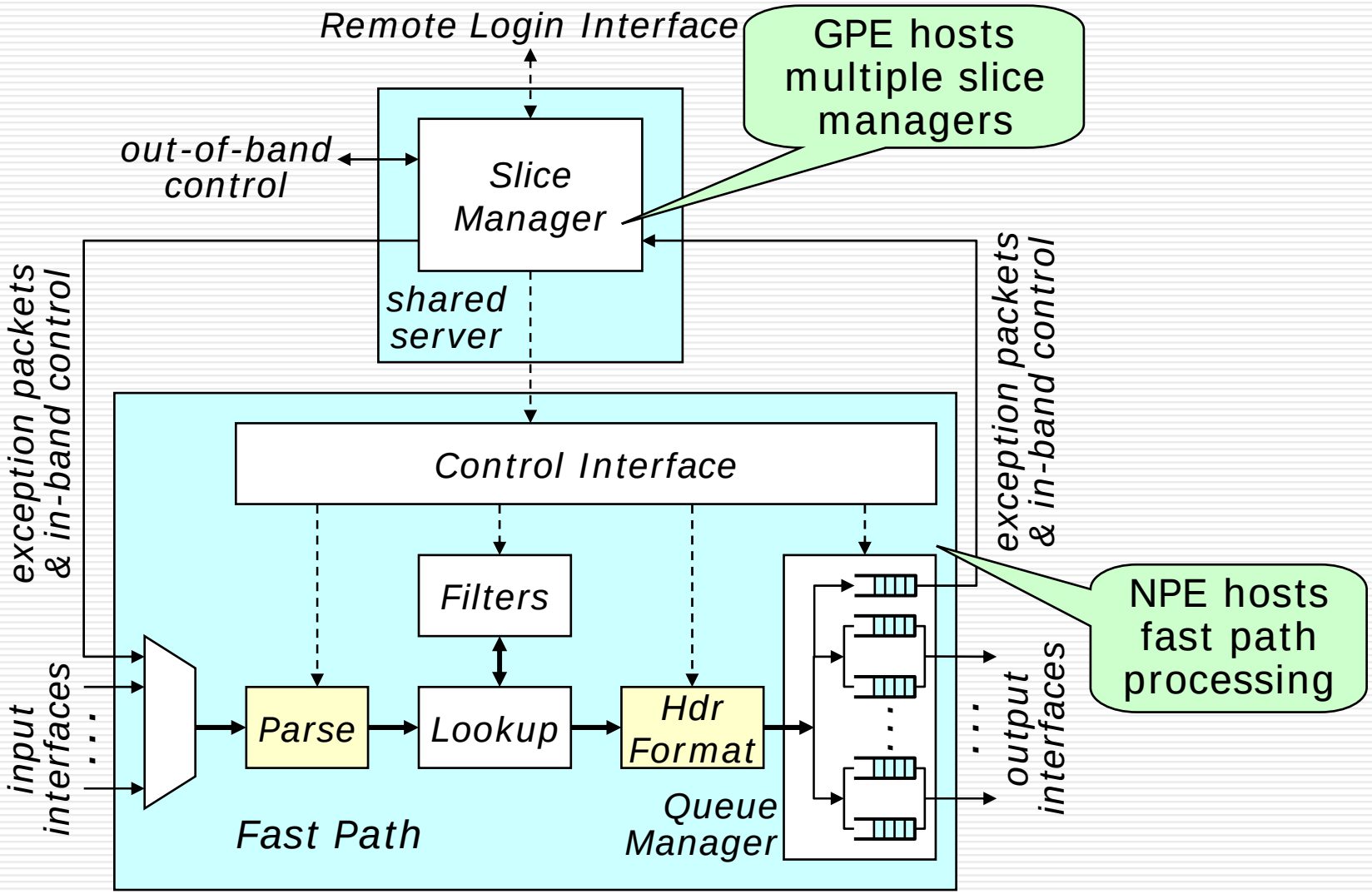
Pipelining & Multi-threading

- Limited program store per ME
  - »parallelize by dividing program among pipeline stages
- Use multi-threading to hide memory latency
  - »high latency to off-chip memory (>100 cycles)
  - »modest locality of reference in net workloads
  - »interleave memory accesses to keep processor busy

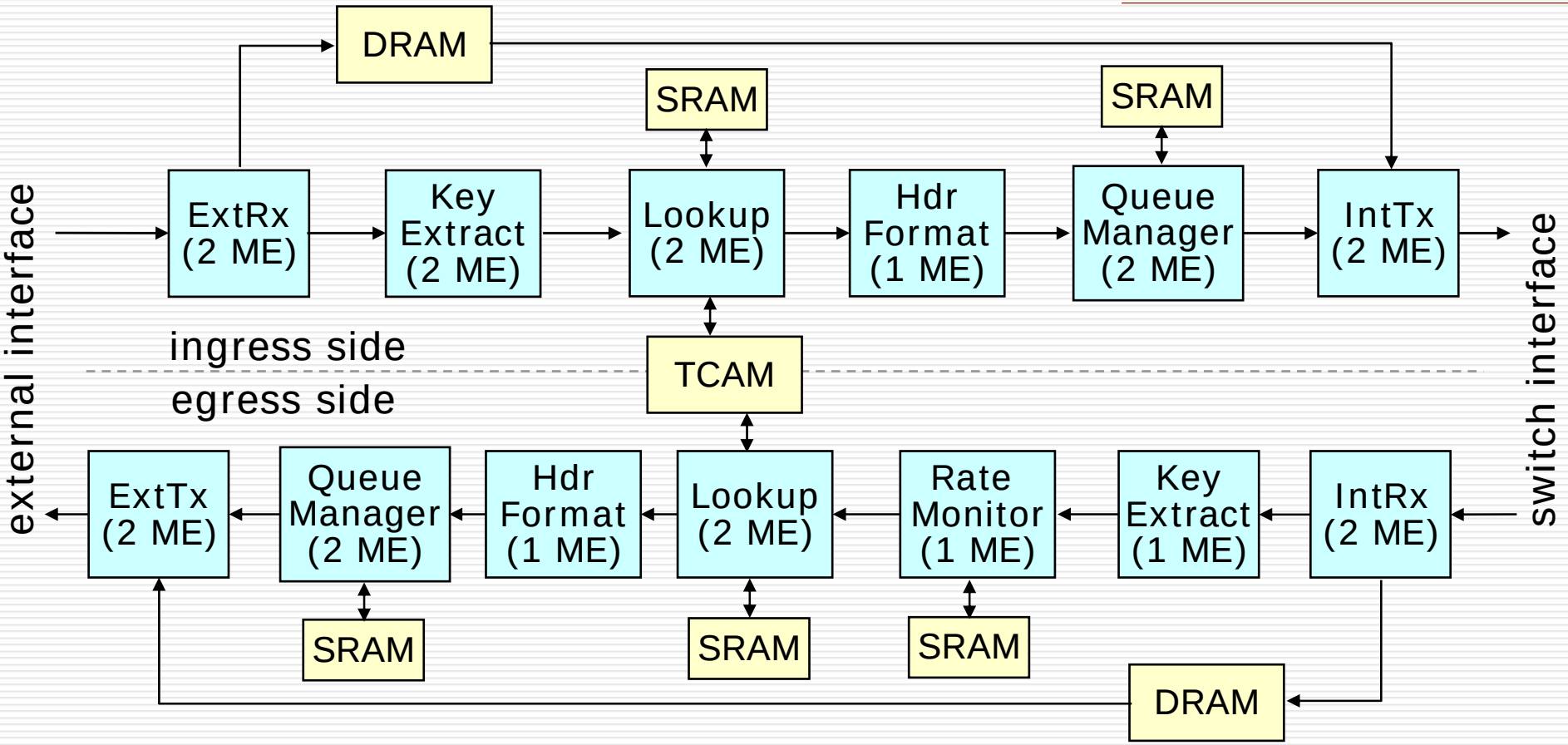


  - »sequenced “hand-offs” between threads maintains order
    - works well when one limited processing time variation

Fast Path/Slow Path App. Structure

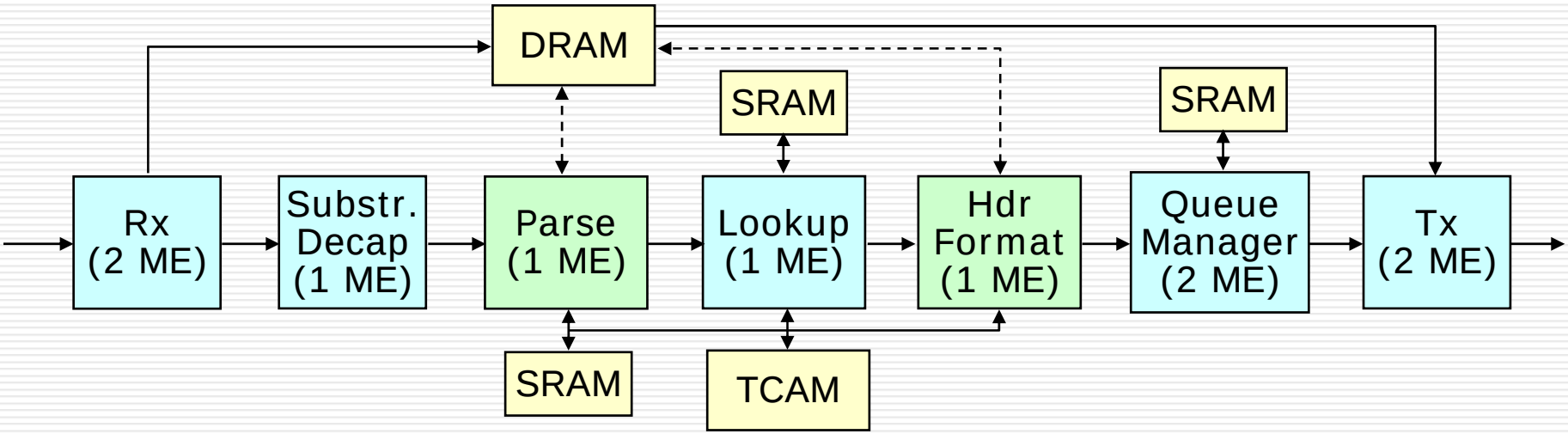


Line Card



- Ingress side demuxes with TCAM filters (port #s)
- Egress side provides traffic isolation per interface
- Target 10 Gb/s line rate for 80 byte packets

NPE Hosting Multiple Apps

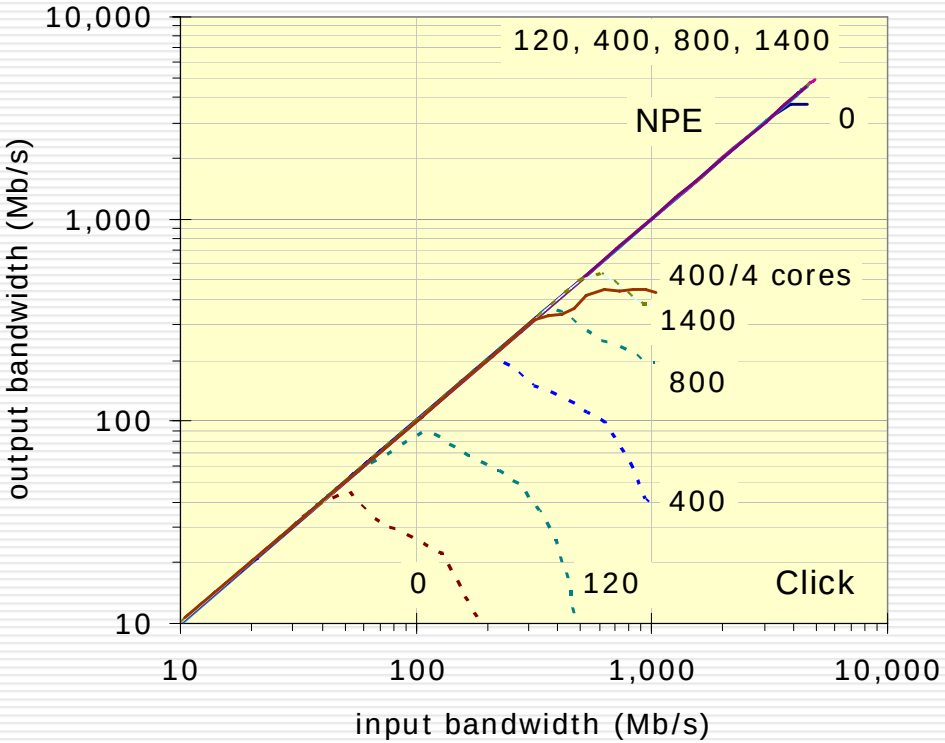


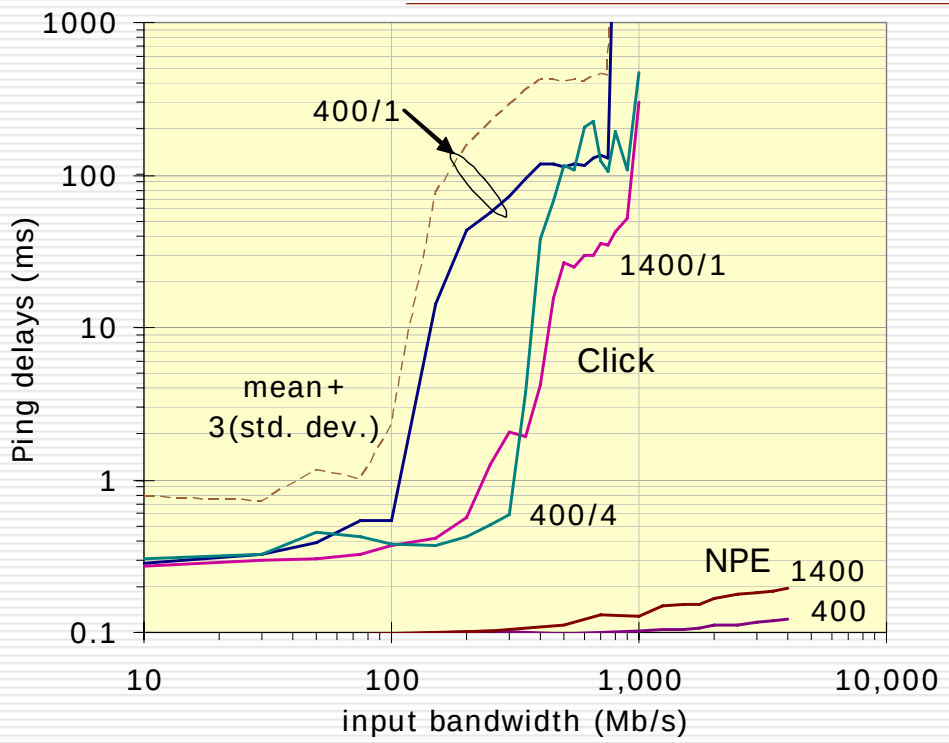
- Parse and Header Format include slice-specific code
  - »parse extracts header fields to form lookup key
  - »Hdr Format makes required changes to header fields
- Lookup block uses opaque key for TCAM lookup and returns opaque result for use by Hdr Format
- Multiple static code options can be supported
  - »multiple slices per code option
  - »each has own filters, queues and block of private memory

Early Application Experience

- IPv4 router fast path
  - packets enter/leave thru UDP tunnels
  - routes/filters in TCAM determine forwarding behavior
  - exception packets forwarded to slice manager on GPE
  - manually configured at this point – plan to add XORP later
- Internet Indirection Infrastructure (I3)
  - explores use of indirection as key network service
  - packets sent to “triggers” which can redirect them to destination
  - uses Distributed Hash Table techniques (specifically, Chord)
  - fast path processing
    - if current node does not store trigger, send to next hop using Chord routing table
    - if current node does store trigger and it’s a simple case, process and forward packet to destination
    - otherwise, send to slice manager on GPE
- Performance compared to conventional Plab case

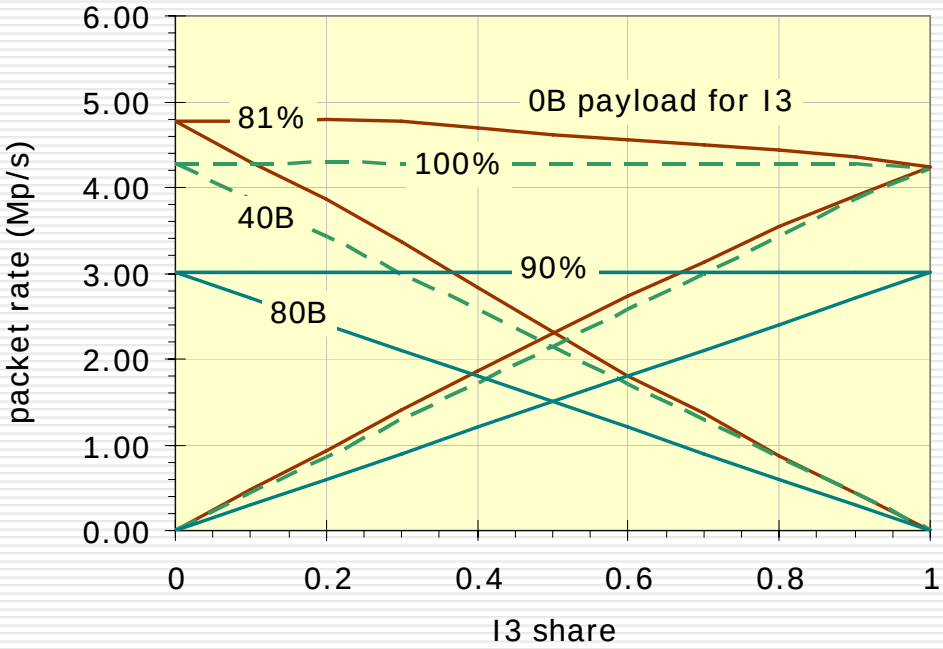
IPv4 Performance Comparisons

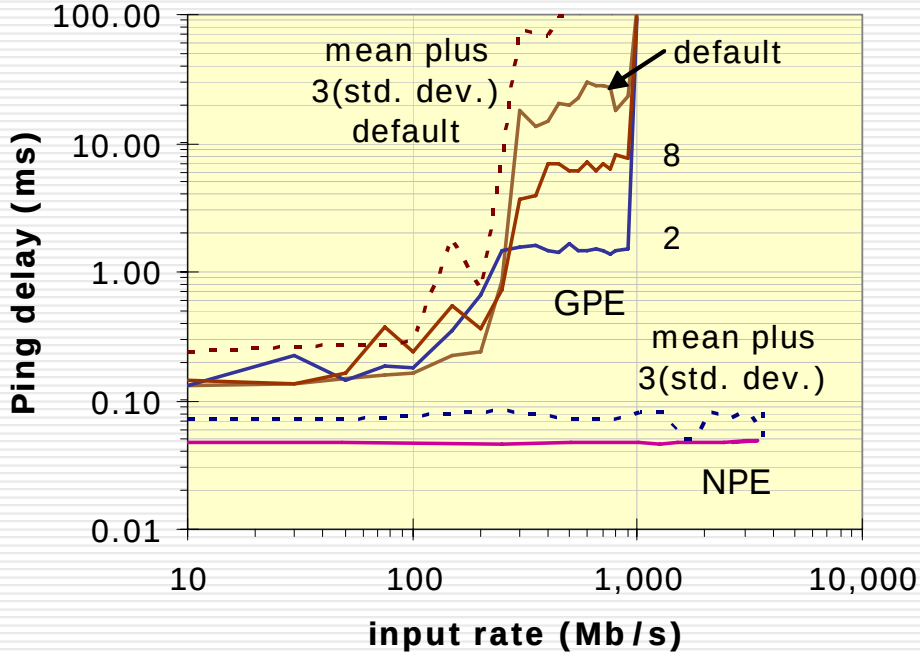




- NPE hosting IPv4 forwarding application
- GPE hosting Click software router in user-space
- 80x improvement in packet processing rate
- Over 100x latency improvement at high loads
  - » measuring ping delays in presence of background traffic

I3 Performance





- Throughput for mix of I3 and IPv4 on NPE
  - »for shortest packets, max I3 rate lower than for IPv4
  - »unexpectedly low performance for 80B payload (fragmentation)
- Latency comparison (using separate slice for ping traffic)
  - »average NPE ping delay under 50 μs (round-trip)
  - »adjusting Plab scheduling parameters improves GPE latency

Things to do Prior to Deployment

- Equipment on order for two systems (6 slot chasses)
  - still need to secure donation from Intel for 4 server blades
- Datapath
  - logging of outgoing traffic
  - adding NAT and ARP to Line Card
  - improving performance (deferrable)
  - next version NPE code to use both NPs
- Control
  - Global/Local Node Managers
    - download slice descriptions from PLC, configure slices on GPEs
  - Global/Local Resource Managers
    - allocation of NPE resources
    - allocation of external port numbers
  - Login session manager
    - login to CP with application-level redirection to GPE

Overview

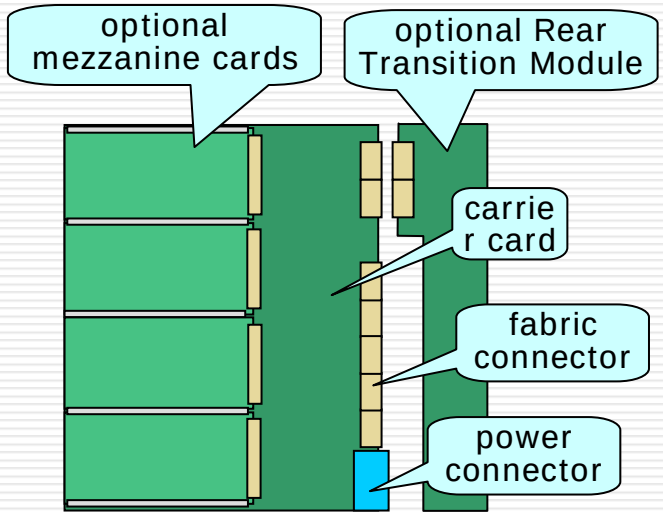
- High performance platform for overlay hosting services
  - » current focus on PlanetLab
  - » hardware is fully assembled and operational
  - » developed method to allow slices to share NP blades
  - » order-of-magnitude performance gains for two applications
- Proposed design for GENI backbone platform
  - » overall architecture and realization using ATCA
  - » potential for scaling it up
  - » how it might relate to flexible optical layer
  - » missing pieces
  - » things to be done
- Open Network Lab and GENI
- Thoughts about requirements
  - » balancing desires/needs with what's feasible/sensible

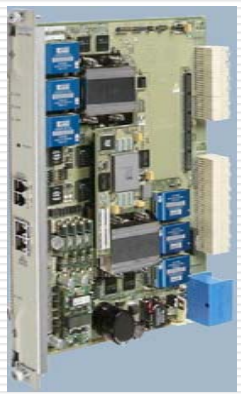
General Approach

- Avoid reinventing the wheel
  - leverage industry standards and trends
  - use available products wherever possible
  - partner with companies to help fill blanks
- Architectural neutrality
  - avoid favoring one style of network over another
  - variety of protocols, service models – don't play favorites
  - minimize functions placed in "substrate"
    - substrate will be difficult to change
    - anything we might want to change belongs in experimental networks, not in substrate
- Flexible use of resources
  - give users "raw" access to resources
  - support variety of resource types
  - stay open to addition of new resource types

Advanced Telecom Computing Architecture

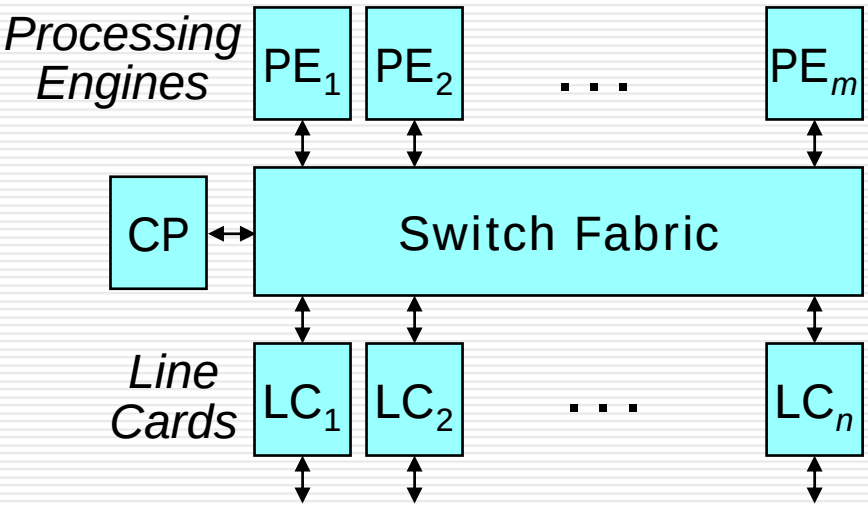
- ATCA is new industry standard
  - » hundreds of companies
  - » defines standard packaging system
  - » enables assembly of systems using components from different suppliers
- Standard 14 slot chassis
  - » high bandwidth serial links
  - » support variety of processing blades
  - » connections to redundant switch blades
  - » integrated management features
- Relevance to GENI
  - » enables flexible, open subsystems
  - » compelling research platform
  - » faster transition of research ideas





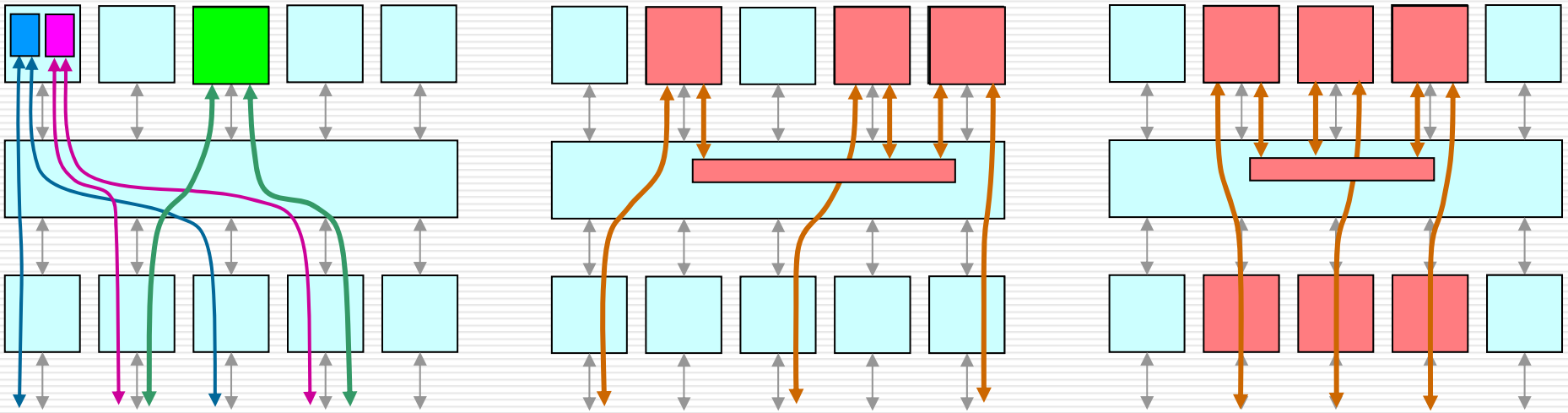
Processing Pool Architecture

- Processing Engines (PE)
  - » used to implement metarouters
  - » variety of types
    - general purpose (GPE)
    - network processor (NPE)
    - FPGA-based (FPE)
  - » raw mode or cooked
    - cooked mode allows shared use
    - includes substrate functionality



- Line Cards and switch fabric belong to substrate
  - » mux/demux packets on external links
  - » move packets to/from PEs
  - » provide isolation among metarouters
  - » allow unconstrained traffic flow within metarouters
    - PEs within MRs manage internal congestion

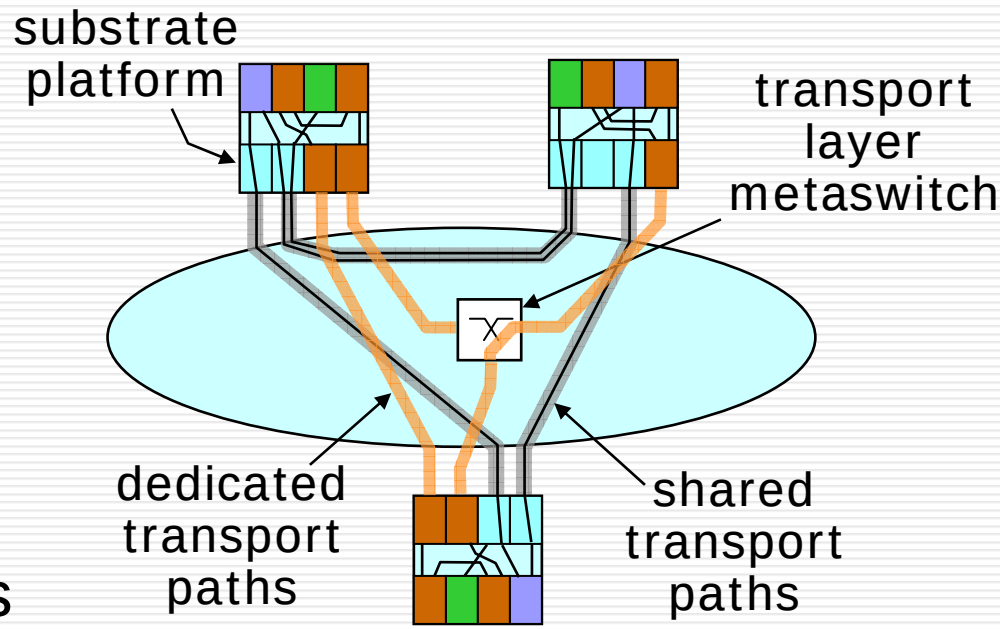
Usage Scenarios



- Multiple MRs sharing single PE
  - » typical case: GPE with MRs implemented as vServers
- Single MR using entire PE
  - » allows MR complete control over PE
    - substrate must rely on switch and LCs for isolation
- High throughput MRs using multiple PEs
  - » may include dedicated LCs
    - allows complete control over transmission format

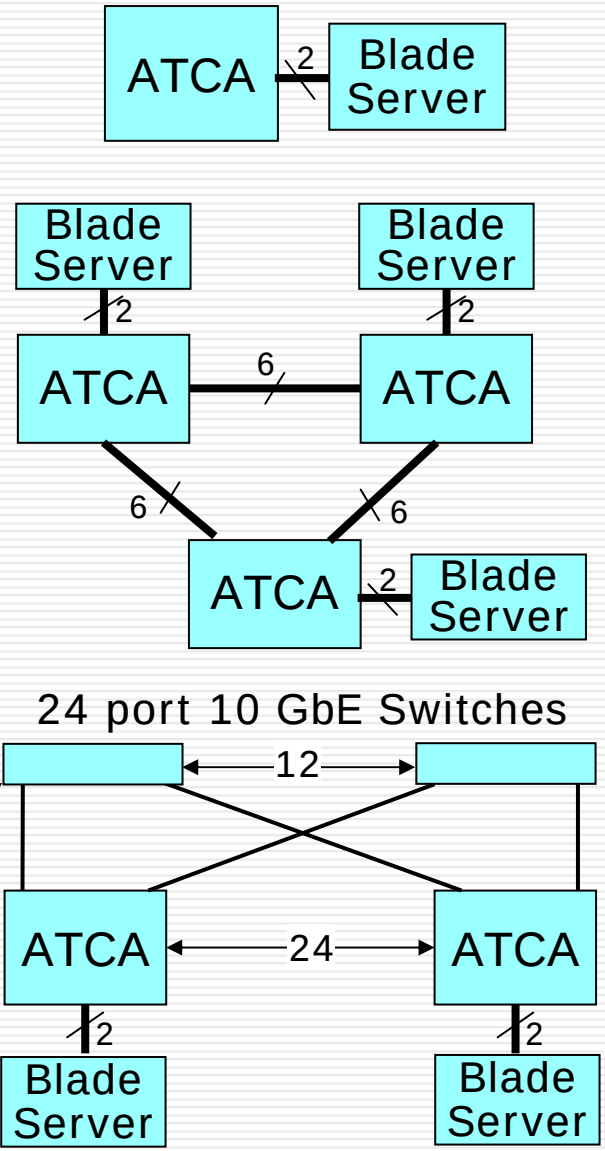
Flexible Optical Transport Layer

- Shared transport paths carry multiple metalinks
  - »semi-static transport level configuration
  - »dynamic configuration of metalinks within path
- Dedicated transport paths serve larger metanets
- Transport layer metaswitches give metanets more direct control over transport layer resources
  - »allows metanet to shift capacity as traffic changes
  - »may be implemented using a shared cross-connect
  - »potentially highly dynamic

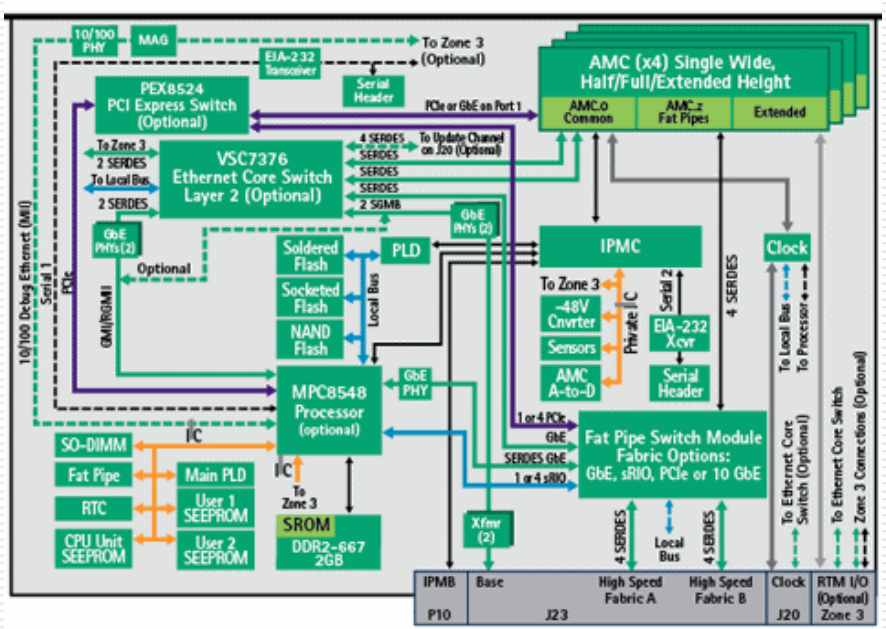


Scaling Up

- Baseline configuration (1+1)
  - » single ATCA plus GP blade server
  - » 14 GPEs + 9 NPEs+ 3 LCs
  - » 10 GE inter-chassis connection
- Multi-chassis direct
  - » up to seven chassis pairs
  - » 98 GPEs + 63 NPEs+ 21 LCs
  - » 2-hop forwarding as needed
- Multi-chassis indirect
  - » up to 24 chassis pairs
  - » 336 GPEs + 216 NPEs + 72 LCs



Adding FPGA Packet Processors



- Carrier card with 10 GbE switch and 4 AMC slots
- AMC with high end FPGA plus DRAM and SRAM
  - »remotely programmable
  - »expansion area could be used to add TCAM
- Alternate approach using non-ATCA cards with 1 GbE or 10 GbE interfaces

Missing Hardware Components

- FPGA-based Processing Engine
  - required components may be available
- Line Card suitable for optical networking experiments
  - NP-based LC requires different RTM at least
  - FPGA-based LC may be more suitable
    - might be able to leverage FPE
    - requires suitable RTM
- Circuit-based fabric switch
  - not strictly necessary – can always carry “TDM frames” across switch 10 GbE switch inside Ethernet frames
  - but, for simple circuit pass-through, may be easy to add to an existing switch chip
    - not clear if switch chip vendor will consider it worthwhile
    - on the other hand, may already be present as “hidden feature”

Things to Be Done for GENI

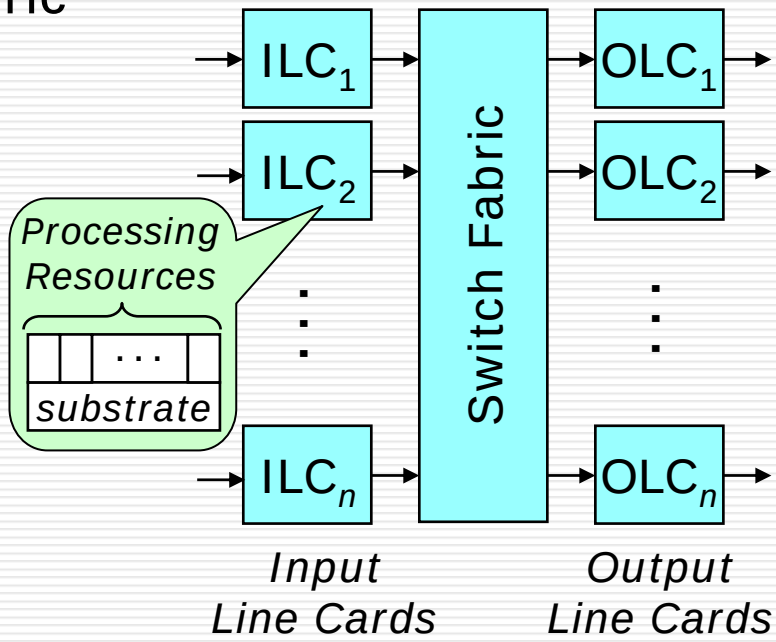
- All essential hardware components available now
  - full ATCA chassis (2 switches, 3 LCs, 9 NPEs) costs ≈$150K
  - full blade server chassis (14 GPEs) costs ≈$65K
  - 24 port 10 GbE switch costs ≈$20K
- Develop missing pieces – needs further investigation
  - FPE, LC for optical networking, circuit-based switch
- Software development
  - view SPP code as version 0 – expect to re-write most of it
  - system-wide configuration management
    - equipment discovery & bootstrapping; configuring slices/
  - NPE datapath software
  - NPE management software
  - NPE sample applications and toolkits
- FPE support software and logic
  - loading and initializing bitfiles; sample applications and toolkits

Alternative Approaches

- Multicore server blades vs. NP blades
  - » quad-core Intel, eight-core Sun chips available now
  - » larger market means more current technology
  - » to come close to NP performance, must program for parallel execution and move fast path down to limit overheads
    - same trade-off of performance vs. developer-friendliness
- Component-based implementation
  - » cluster of rack-mount servers connected by 10 GbE switch
  - » other components (IO cards, NP servers, FPGA servers) also connect to 10 GbE switch
  - » separate 1 GbE control switch to keep system management traffic separate from data traffic
  - » architecturally no different from ATCA approach
    - just less integrated, less compact and requiring more power
    - may be less expensive to purchase, but likely more expensive to operate and manage

Virtualized Line Card Architecture

- Similar to conventional router architecture
  - » line cards connected by a switch fabric
  - » traffic makes a single pass through the switch fabric
- Requires *fine-grained virtualization*
  - » line cards must support multiple *meta line cards*
  - » requires intra-component resource sharing and traffic isolation

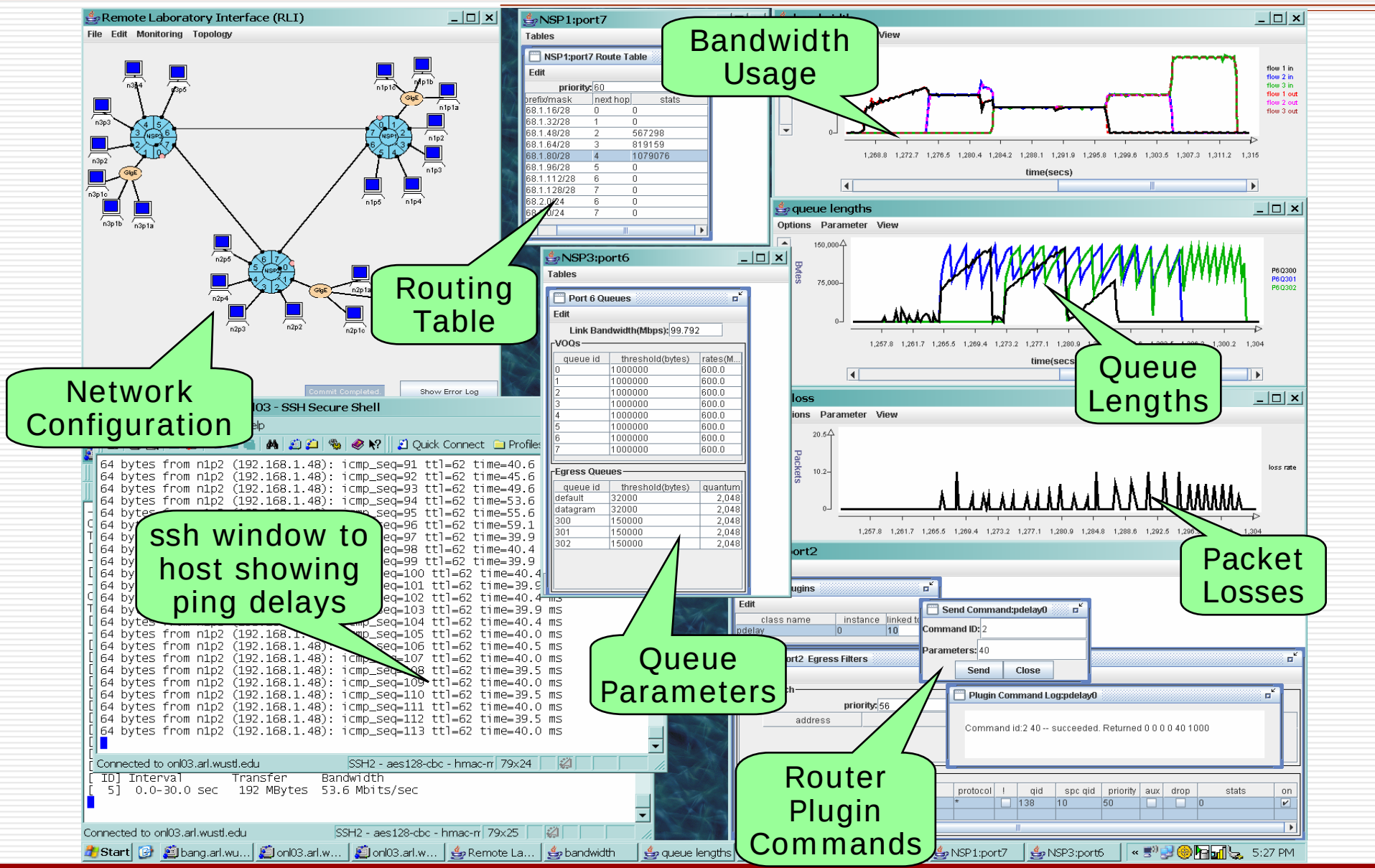


- Mismatch for current device technologies
  - » multi-core NPs lack memory protection mechanisms
  - » lack of tools and protection mechanisms for independent, partial FPGA designs
- Hard to vary ratio of processing to IO

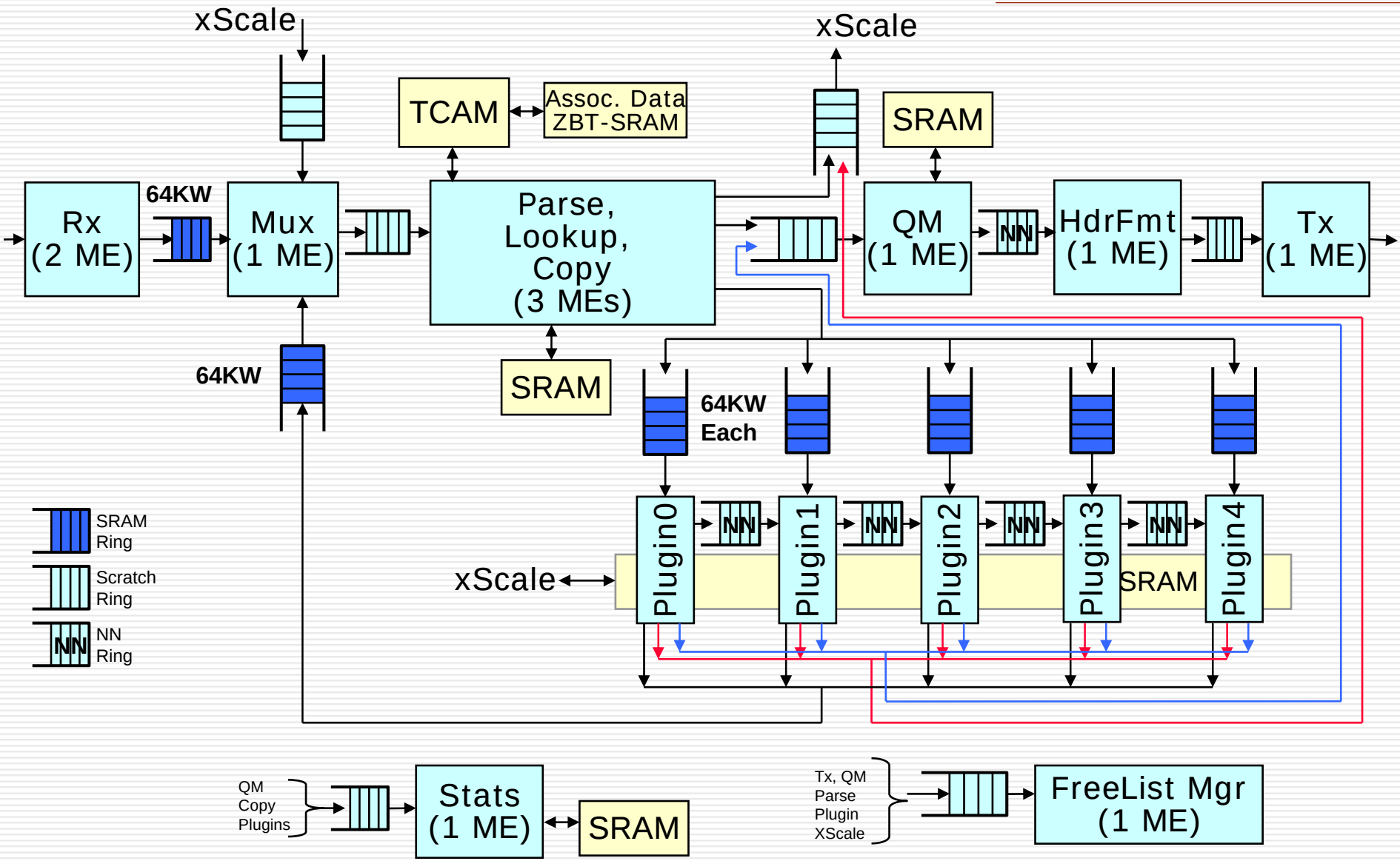
Open Network Lab and GENI

- ONL is Internet-accessible networking lab
  - »built around set of extensible gigabit routers
  - »intuitive Remote Lab Interface makes it easy to get started
  - »extensive facilities for performance monitoring
- Expansion underway
  - »14 new Network Processor (NP) based routers
    - packet processing implemented in software for greater flexibility
    - high performance plugin subsystem for user-added features
    - support larger experiments and more concurrent users
  - »70 new rack-mount computers to serve as end systems
  - »4 stackable 48 port GbE switches for configuring experiments
- Staging area for GENI
  - »“gentle” environment for experimenting with NP-based routers
  - »plugin mechanism enables full range of user-defined extensions
  - »fewer constraints on NP use than in SPP context

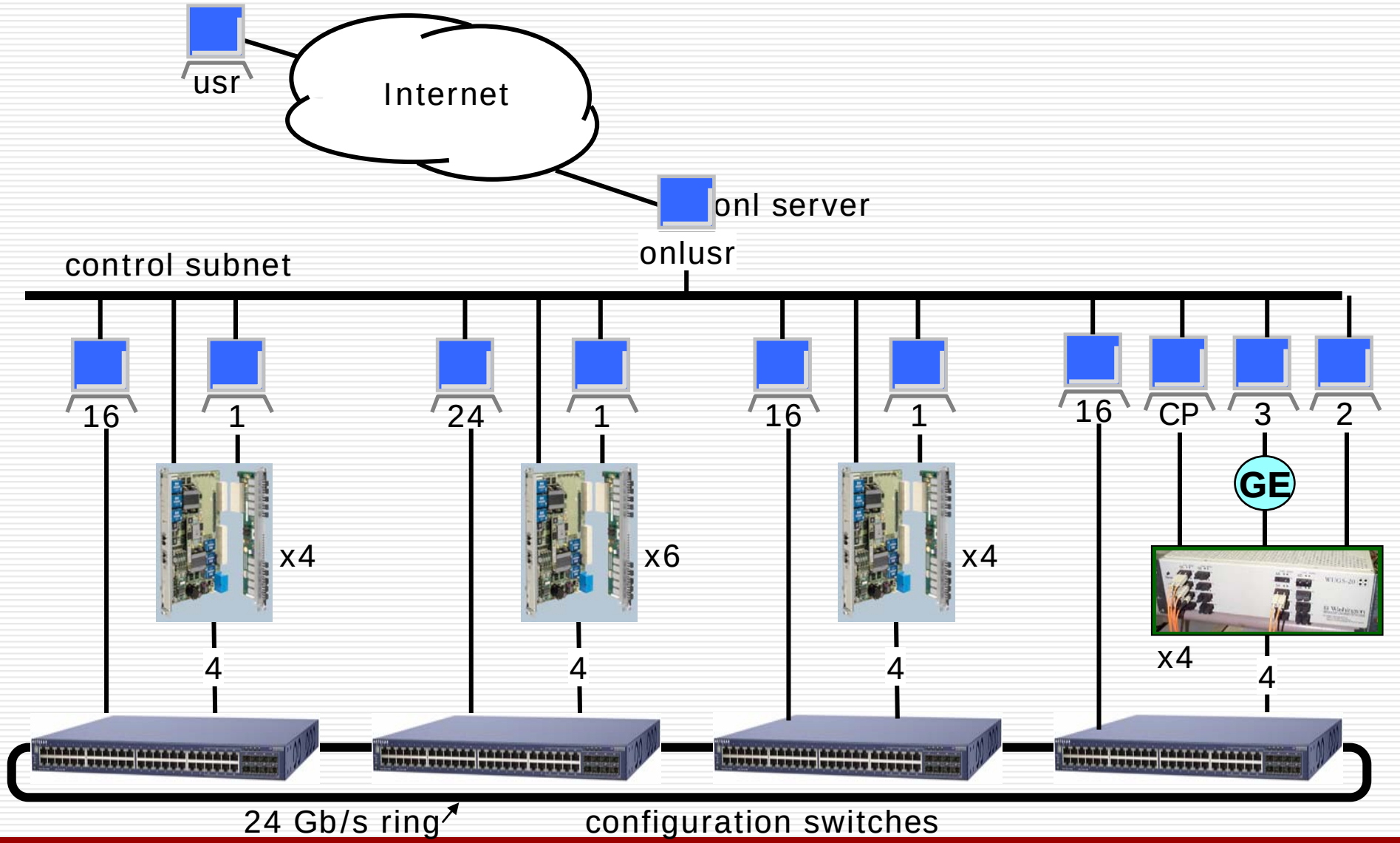
Sample ONL Session



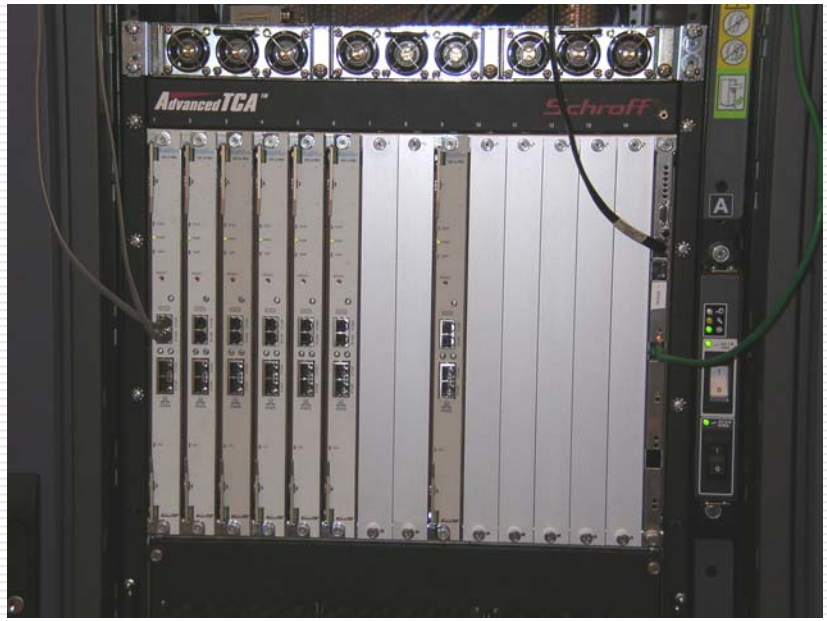
ONL NP Router



Expanded ONL Configuration



Equipment Photos



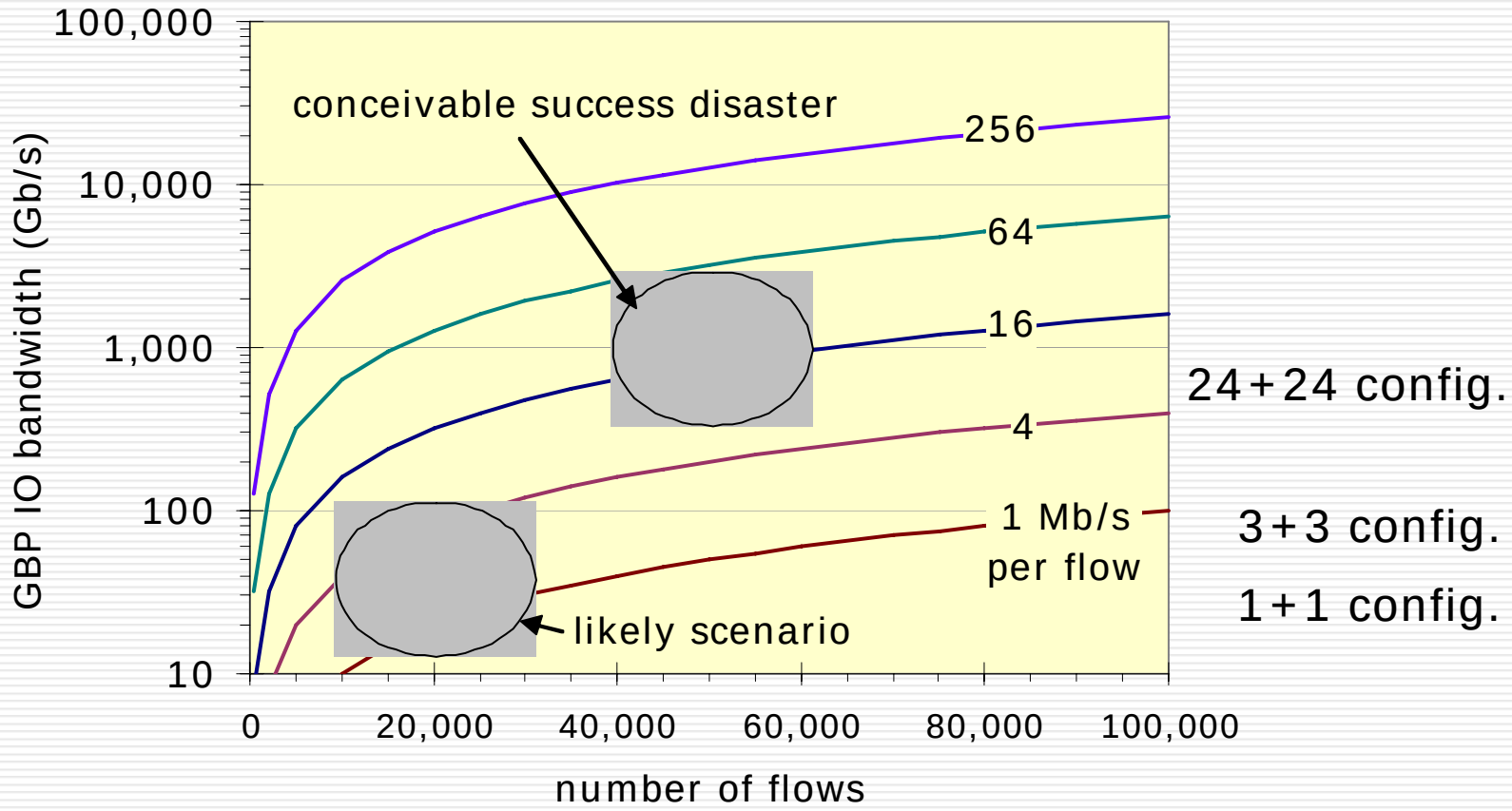
Overview

- High performance platform for overlay hosting services
  - » current focus on PlanetLab
  - » hardware is fully assembled and operational
  - » developed method to allow slices to share NP blades
  - » order-of-magnitude performance gains for two applications
- Proposed design for GENI backbone platform
  - » overall architecture and realization using ATCA
  - » potential for scaling it up
  - » how it might relate to flexible optical layer
  - » missing pieces
  - » things to be done
- Open Network Lab and GENI
- Thoughts about requirements
  - » balancing desires/needs with what's feasible/sensible

High Level Objectives

- Enable experimental networks and minimize obstacles
  - »provide resources and stay out of the way
  - »architectural neutrality
  - »enable use by real end users
- Stability and reliability
  - »reliable core platform
  - »effective isolation of experimental networks
- Ease of use
  - »enable researchers to be productive without heroic efforts
  - »toolkits that facilitate use of high performance components
- Scalable performance
  - »enable GENI to support >100K users
  - »metarouters with wide range of capacities (1000x range)
- Technology diversity and adaptability
  - »variety of processing resources
  - »ability to add new resource types as they become available

How Big Need it Be?



- For 100K users, GBP should support 20K flows
  - »but what average bandwidth per flow?
- Enable high ratio of processing to IO
  - »at least 3x the processing of conventional routers

My Research Drivers for GENI

- GENI as prototype for next-generation Internet
  - »commercial overlay hosting service model
  - »diversified Internet model
    - multi-domain substrate that provides resources and user connections
    - metanetworks that span multiple substrate domains
- Novel network services
  - »location-aware network addressing/routing
  - »network-embedded media processing
  - »informed update distribution for DIS, virtual worlds
- Applications that can leverage advanced net services
  - »virtual presence applications
  - »using virtual worlds to enable real-world collaboration
    - improve organizational productivity & replace much physical travel with virtual travel to help meet global environment/energy challenges
  - »interacting with the real world
    - remote operation of scientific instruments (telescopes, microscopes, network testbeds), infrastructure systems, telerobotics

Possible Next Steps

- Deploy SPP nodes in PlanetLab so others can use
  - » at this point, limited to two nodes
    - could add more at ≈$60K per node
  - » add dedicated 1 GbE links among SPP nodes (VINI?)
  - » need more resources
    - for managing the development effort
    - for implementing control software
    - for developing user documentation
    - for using the system and driving improvements
- Assemble larger systems and start filling gaps
  - » 14 slot chassis coming soon
  - » should start effort on multi-chassis configurations
  - » FPGA-based processing engine
  - » Line Card for direct access to optical layer