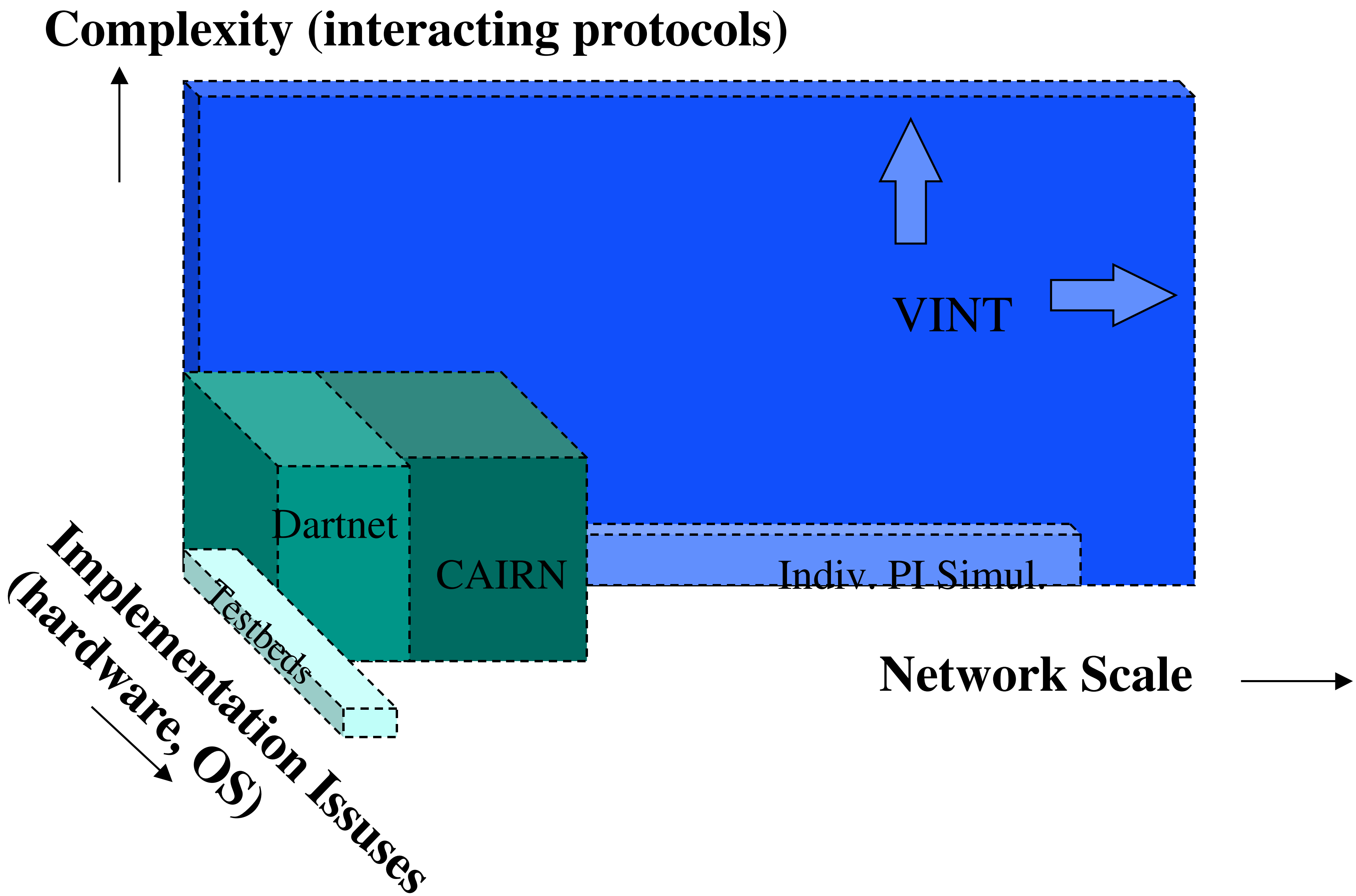
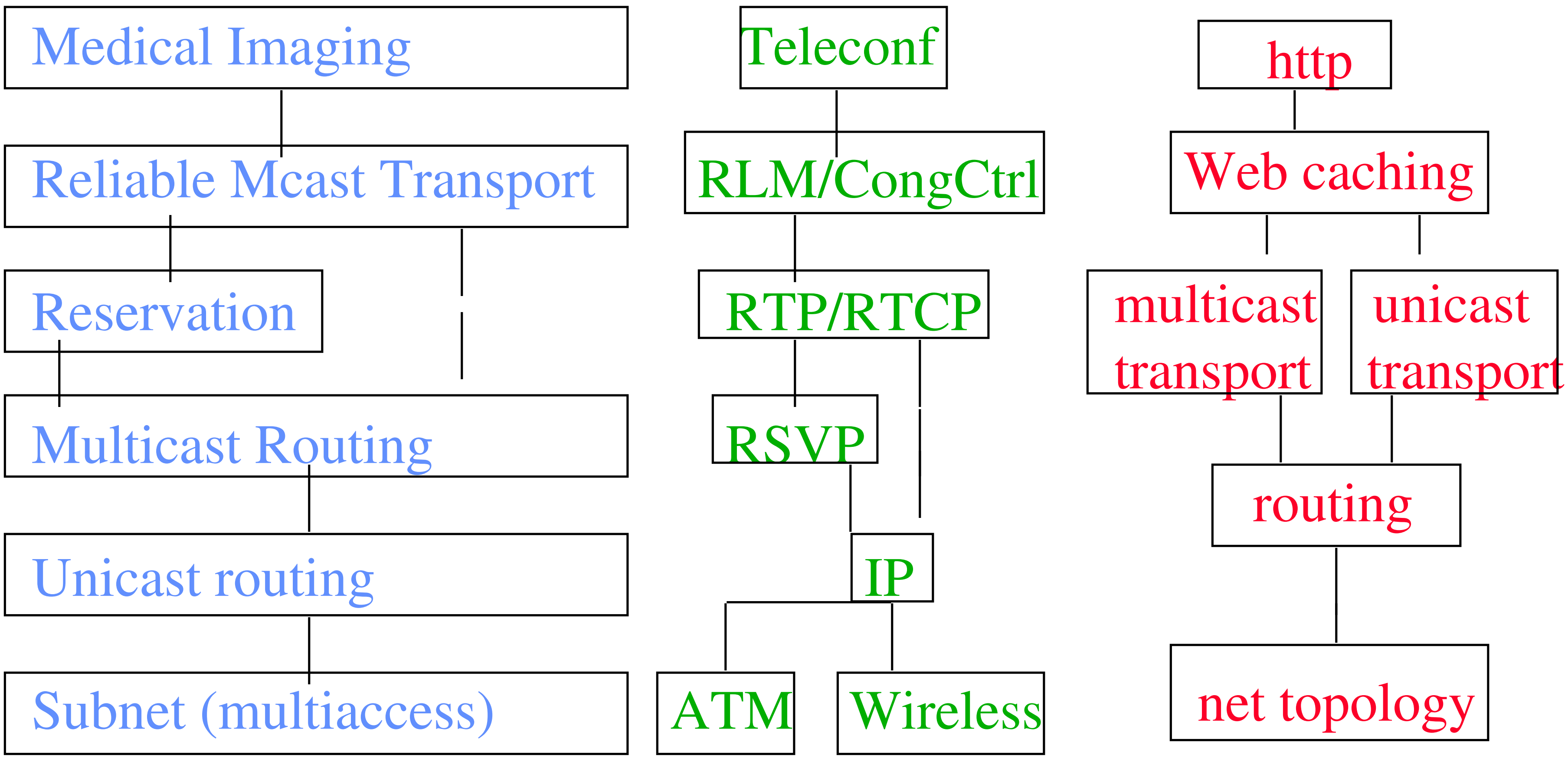
Virtual InterNet Testbed (VINT)

- Participants:
  - USC/ISI: Deborah Estrin (*), Mark Handley, John Heideman, Ahmed Helmy, Polly Huang, Satish Kumar, Kannan Varadhan, Daniel Zappala
  - LBNL: Kevin Fall, Sally Floyd (*), Van Jacobson
  - UCBerkeley: Elan Amir, Steve McCanne (*)
  - Xerox PARC: Lee Breslau (*), Scott Shenker (*)
  - other ns users

  (*) Co-PI
- Funded under BAA 96-06.


Complexity (interacting protocols)
VINT
Dartnet
CAIRN
Indiv. PI Simul.
Testbeds
Network Scale
Implementation Issues
(hardware, OS)

Example Stacks of Interacting Protocols



Today's Practices

- Individual simulators used in Protocol Research and Design are:
  - **narrow**: simulation focuses on single protocol
  - **isolated**: massive duplication of effort across PIs, lack of common test suites
  - **primitive**: minimal visualization; inadequate load, topology, membership models; scale
- Many critical system/network design issues relate to scale, interaction among components, robustness

Evaluate at intended scale--orders of magnitude larger than testbeds

- Congestion control
  - TCP dynamics with complex traffic mixes and paths
- Integrated services
  - admission control for aggregated flows
  - end-to-end performance across integrated services networks
- Multicast transport
  - reliable multicast feedback/recovery w/large groups
- Routing
  - multicast routing for large topologies
  - tag switching protocols across large topologies

Intended contributions to address scale and systematic methods

- infrastructure to configure simulations
  - libraries of validated network topologies, source models, session and group models
  - scenario generation: loss models, event sequences
- abstraction techniques (session level simulation)
- analysis of complex behavior
- visualization techniques

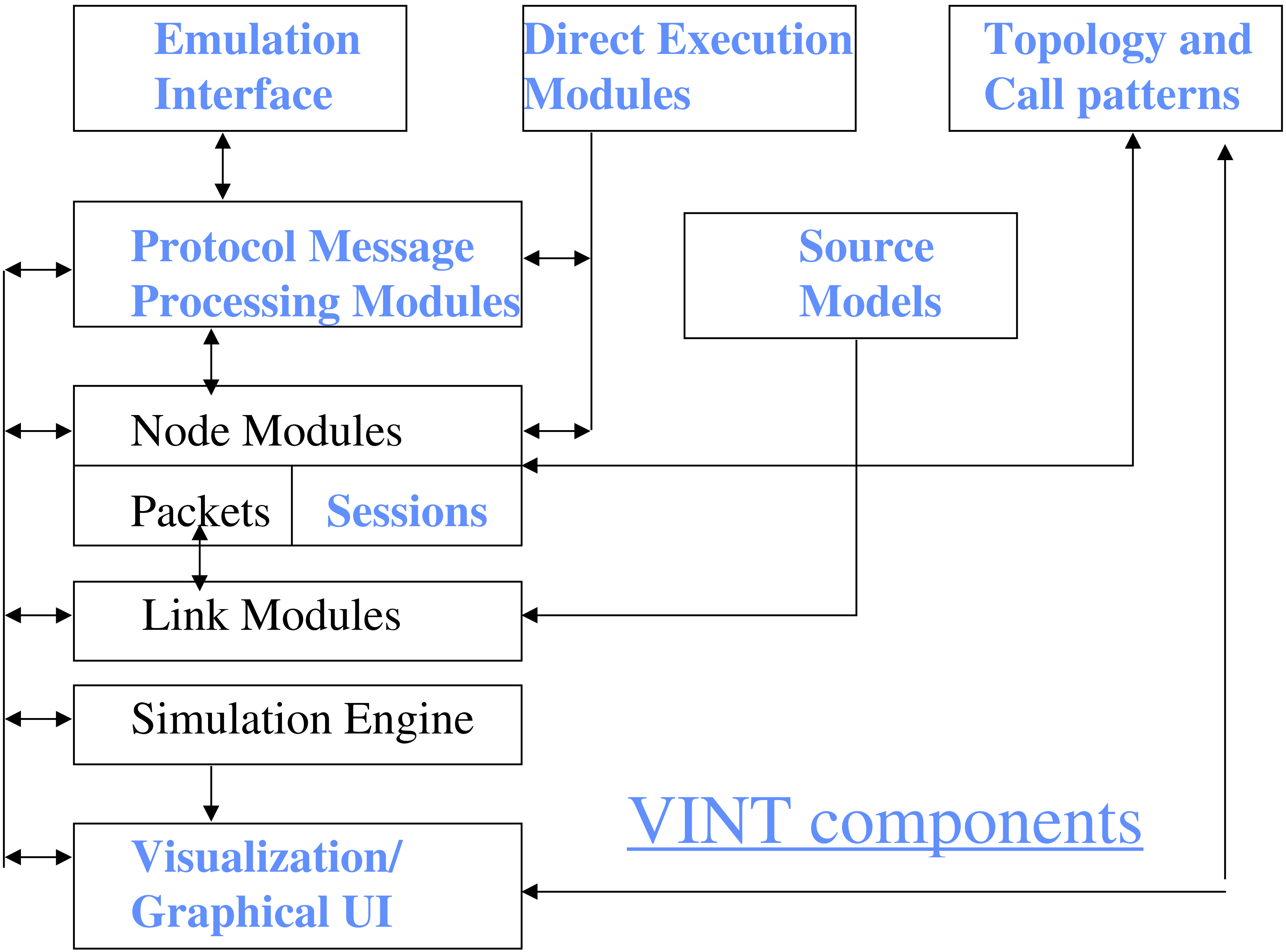
Evaluate designs “in context”--protocol interaction

- reliable multicast protocols with dynamic multicast routing
- heterogeneous congestion control schemes with RED, FQ, other qing strategies and heterogeneous network bandwidths
- multicast routing with dynamic topology, heterogeneous links, and adaptive routing
- web cache management using multicast update
- integrated services for mobile hosts over wireless networks

Intended contributions in protocol interaction

- selective abstraction (single environment for session-level and detailed simulation)
- studies of multiprotocol interaction
- robustness analysis methods, e.g., scenario generation, error analysis

VINT components



Current Projects/Immediate objectives

- Composable simulation framework based on ns (LBNL/UCB network simulator)
- Workload and topology libraries
- Protocol libraries
- Selective protocol abstraction
- Protocol animation

ns (UCBerkeley, LBNL)

- C++ simulation engine, C++ building blocks, OTcl objects, simulation script
- protocol definition and simulation configuration using a rich scripting language (OTcl)
- separates visualization from simulation
- nsv2 deconstructed basic objects to increase flexibility and composability
- modules from multiple contributors mirrors modularity of protocols/development

ns components

Simulation Scripts
Traffic, Topology,
Scenarios, Logging

OTcl objects--basic
components:
router protocols, links,etc.

C++ basic building blocks
forwarding tables, pkt processing
and queuing, scheduling events

Topology and Workload Libraries

- Topology generators
  - GaTech ITM (flat and two-level transit-stub topologies with specifiable properties)
  - ntg (Hotz) for N-level hierarchies with geographic placement
- Traffic generators
  - source models and traffic traces
  - traffic mixes
  - mcast group arrival/departure models
- Common evaluation libraries
  - test suites
  - utility curves

Protocol libraries

- Existing: TCP, Quing (CBQ, FIFO,RED), dynamic unicast routing, simple multicast routing, RLM
- Near completion
  - real time flow monitoring
  - enhanced multicast routing
  - reliable multicast
  - RTP/RTCP
- Planned: RSVP, http, Web caching, ...

Abstraction (session level simulation)

- Abstraction of underlying protocol actions to enhance scalability
- Common user interface and simulation engine allows selective and hybrid abstraction
- Example: reliable multicast with different degrees of underlying protocol abstraction
  - packets sent across centrally computed multicast trees, forwarded hop-by-hop through router queues
  - packets delivered subject to computed delay and loss models

Protocol animation

- enhance nam (LBNL/UCB network animator)
  - rich interface for viewing protocol state in nodes
  - short term scale to moderate size networks (“a screenful”--50 to 100 nodes)
- graphical scenario construction

Longer-term projects/goals

- Techniques for derivation and validation of modules
- Emulation interface (connecting real router/host to simulated network)
- Generalized methods for moving between and combining detailed and abstraction simulations.
- Enhanced visualization/graphical front end
- Emergent-behavior studies (eg. reservation thrashing, congestion control interactions, RLM and SRM dynamics)

VINT Innovative Claims

- Composable component framework based on ns
- Libraries of topologies, workloads, scenarios, protocols
- Abstraction capabilities such as hybrid session and packet level simulation components with common UI and interfaces
- Visualization techniques and tools
- Emulation interface to real implementations
- Scaling studies of key next generation internet design issues

Conclusions

- ***Excited by growing momentum behind use of ns and demand for VINT tools/infrastructure !!!***
- We needed this all yesterday
- see http://netweb.usc.edu/vint