

# **IXP Integration in Emulab ([www.emulab.net](http://www.emulab.net))**

*Abhijeet Joglekar*

Jay Lepreau, Leigh Stoller  
Rob Ricci, Mike Hibler

University of Utah

IXA Education Summit      September 19, 2003

# Emulab Network Testbed

---

- ✍ Time and space-shared reconfigurable cluster, fully Web and ssh-accessible
- ✍ Brings a simulator's ease of use and control to network emulation
  - ✍ Topology creation through *ns* files and a GUI
  - ✍ Automatic mapping of virtual topology consisting of end-nodes, routers and links, to physical resources
  - ✍ Integrated event system can be used to dynamically change link and traffic generator characteristics
  - ✍ Secure remote access to serial console, power control, custom kernels, root access: better than home.
- ✍ Used extensively for network and distributed systems research and education: over 500 users; 4 more sites
- ✍ *Now supports IXP network processors!*

# IXP Integration in Emulab

---

In two ways:

- ✍ Available to remote researchers
  - ✍ Same ease of use and automatic setup of other Emulab resources, now extended to the IXP
  - ✍ Can be configured in an arbitrary network topology with end-nodes and routers, to experiment with and evaluate an IXP-based application
- ✍ As transparent high capacity link-emulators

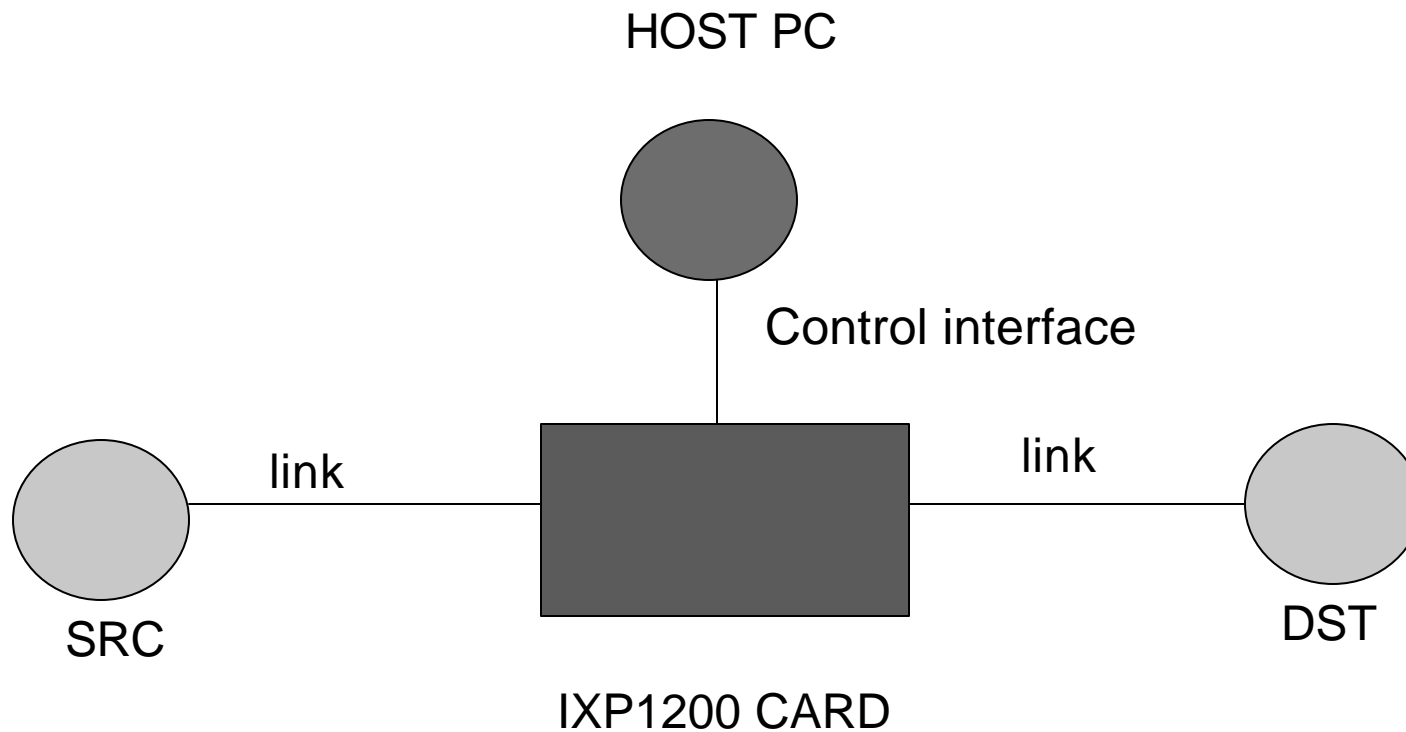
# Challenges

---

- ✍ Issue: an IXP is not 1 computer, it's several!
  - ✍ Host PC
  - ✍ StrongArm
  - ✍ Microengines
  - ✍ Microcode development box? Local or remote?
- ✍ Add a “sub-node” notion throughout Emulab
- ✍ Both IXP and its host are first-class entities
  - ✍ PC can be implicitly allocated and associated to IXP
  - ✍ User gets handles on both so can manipulate them, e.g., load custom image

# Demo Topology

---



# Features

---

- ✍ Routes, interface configuration generated by Emulab.
- ✍ Rutable IP addresses assigned to the control interface between the host and the IXP.
- ✍ Project directories mounted on the card from emulab file server, IXP SDK directory mounted from the host.
- ✍ Hosts, name server files mounted on the card.

# ...Features

---

- ✍ Automatic booting and setup of the card.
- ✍ Serial console, power controller access.
- ✍ Support for a startup command on the IXP, enables download of custom application to the microengines.

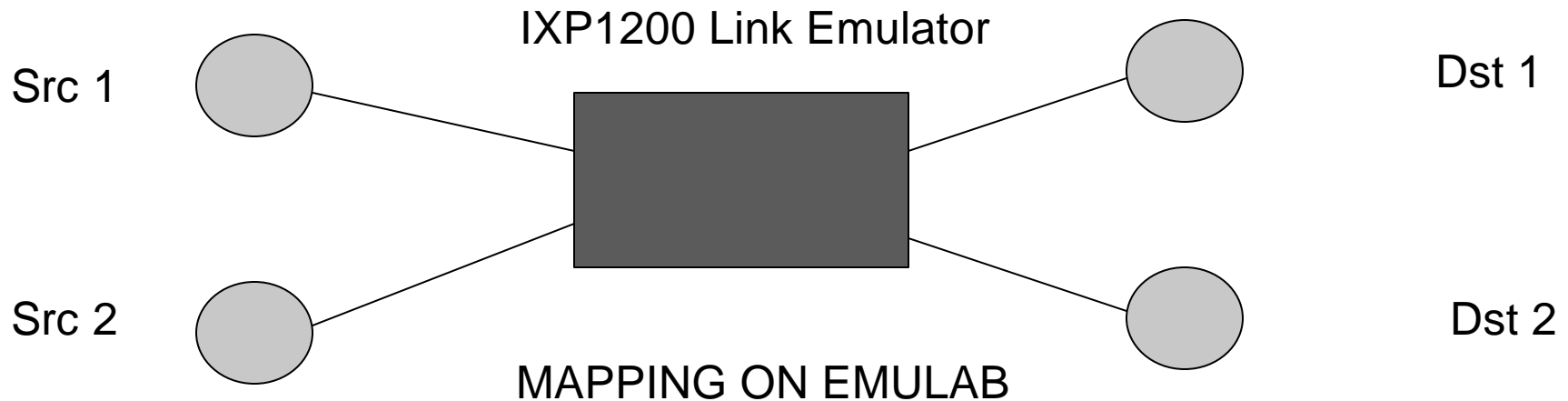
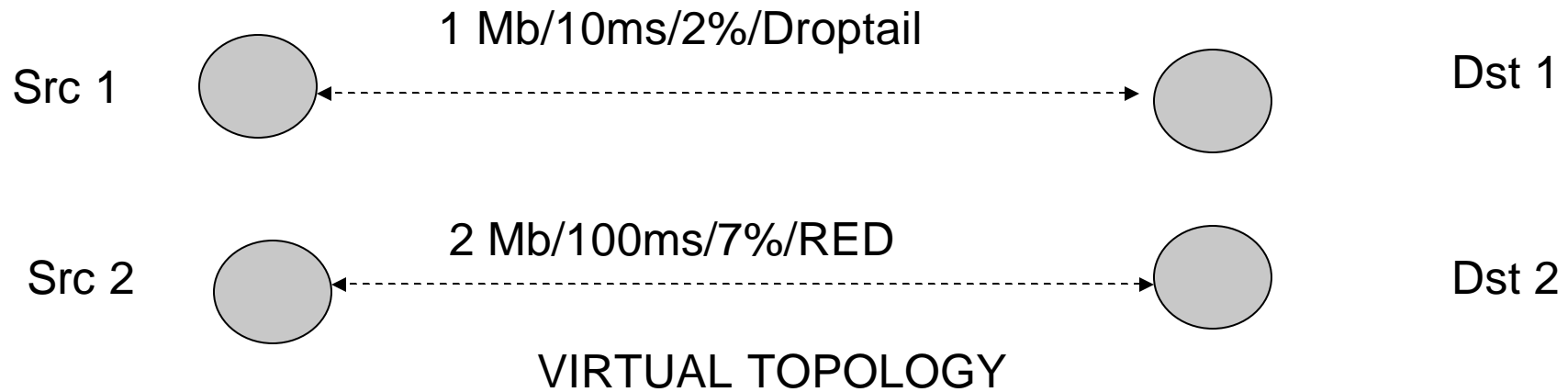
For eg.

*tb-start /proj/foo/myapp /proj/foo/config*

- ✍ Support for windows development environment to be added soon.

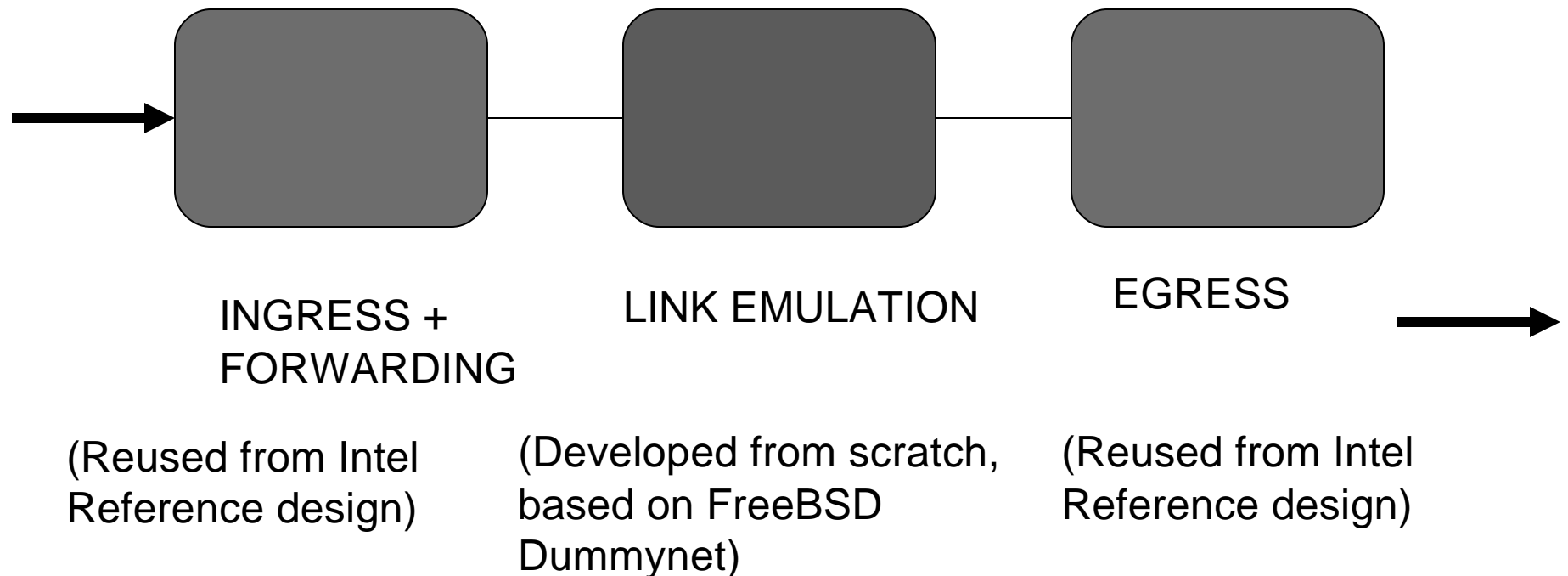
# High capacity link emulation

---



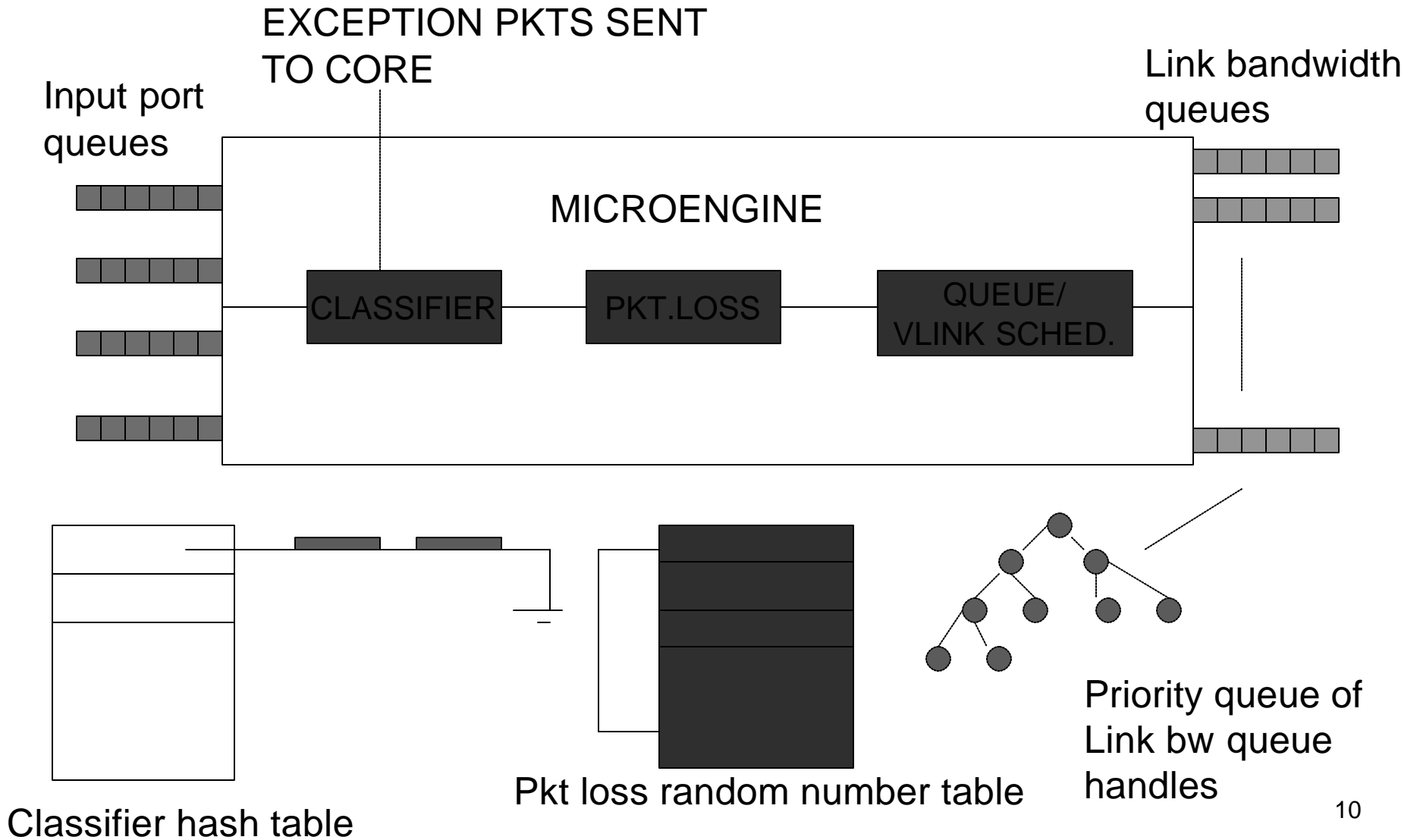
# Architecture I

---



- Link model : bandwidth, latency, queuing, packet loss
- Support for multiplexing links, since typical network experiments use links with low to moderate bandwidths.

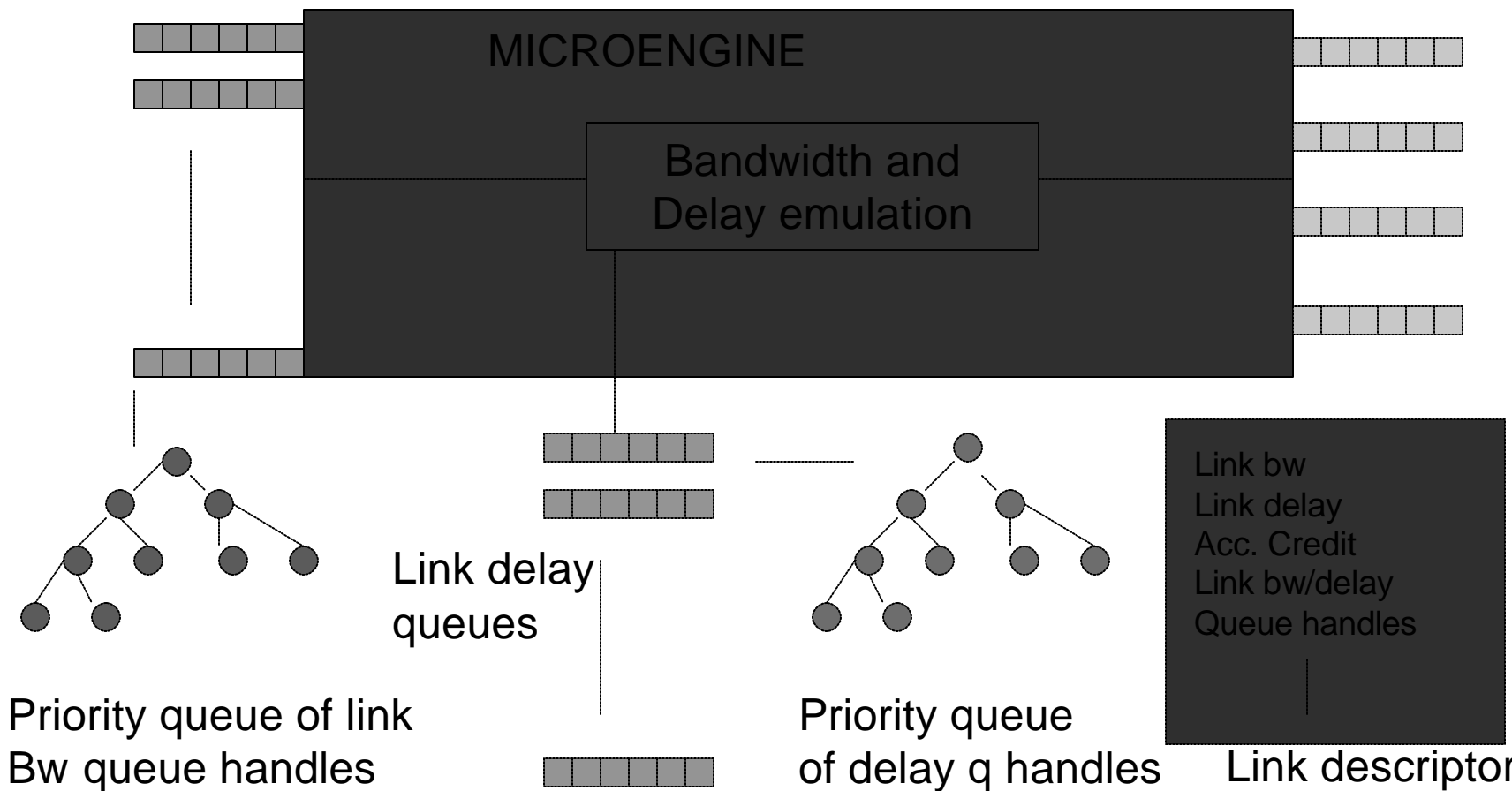
# Architecture II



# Architecture III

Link bw queues

Output port queues



# Preliminary Evaluation

---

## ✍ Validation

✍ Bandwidth: 64 Kbps – 15 Mbps

✍ Delay: 2 – 300 ms

✍ Link loss: 0 – 100 %

## ✍ Forwarding capacity (on 4 slow ports)

✍ Max sized packets: Line rate

✍ 64 byte packets: 83% of line rate

## ✍ Link multiplexing capacity

✍ 180 2Mb links, 120 3Mb links, 72 5Mb links...

# Acknowledgements

---

- ✍ Dirk Brandewie for helping with the initial card setup, and Erik Johnson and Aaron Kunze for discussions on microengine programming.

# Conclusion

---

- ✍ Through IXP deployment in Emulab
  - ✍ Experimentation and evaluation of IXP apps is
    - ✍ Open to a hugely wider community
    - ✍ Made much easier and more realistic (e.g, larger topologies)
    - ✍ Further enriched when new work to support Cisco routers and wireless devices is done
  - ✍ Emulab benefits from high-capacity link shapers