



Evaluation of Advanced Optoelectronic Interconnect Technology

Janice Onanian McMahon

Tom Emberley

MIT Lincoln Laboratory

30 July 2001



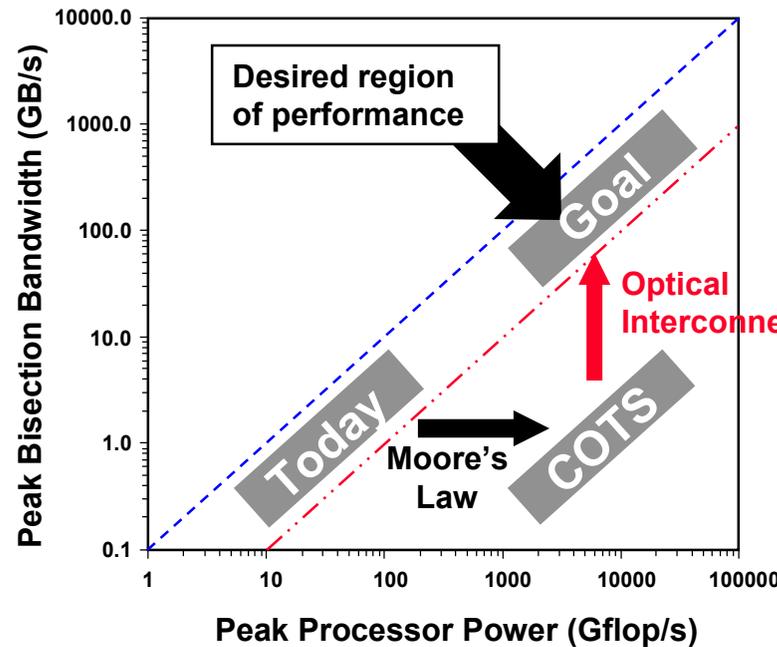
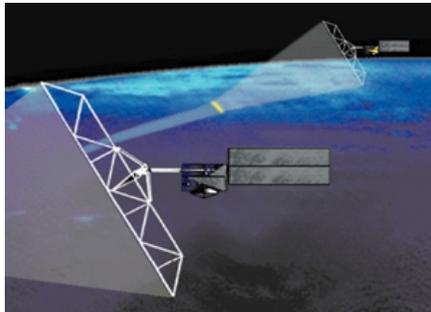
Outline

- **Introduction**
- **Performance Modeling**
- **Simulation Roadmap**
- **Detailed Design**
- **Summary**

- 
- *Program Goals*
 - *Technical Approach*



Real-Time Embedded Signal Processing



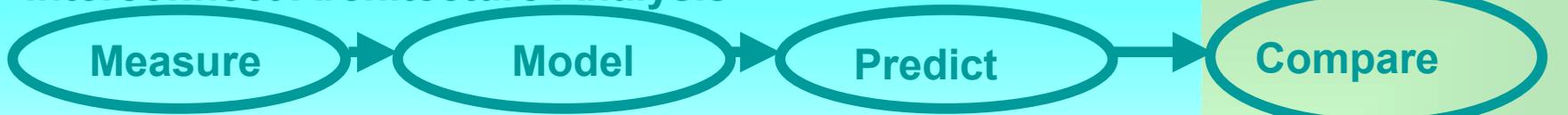
**× 1000
speed/(power·volume)
over electrical based
interconnects**

- Architectural balance is key to achieving delivered performance
- VLSI Photonics will help maintain this balance for these target applications



MIT/LL Program Goals FY99-FY02

Interconnect Architecture Analysis



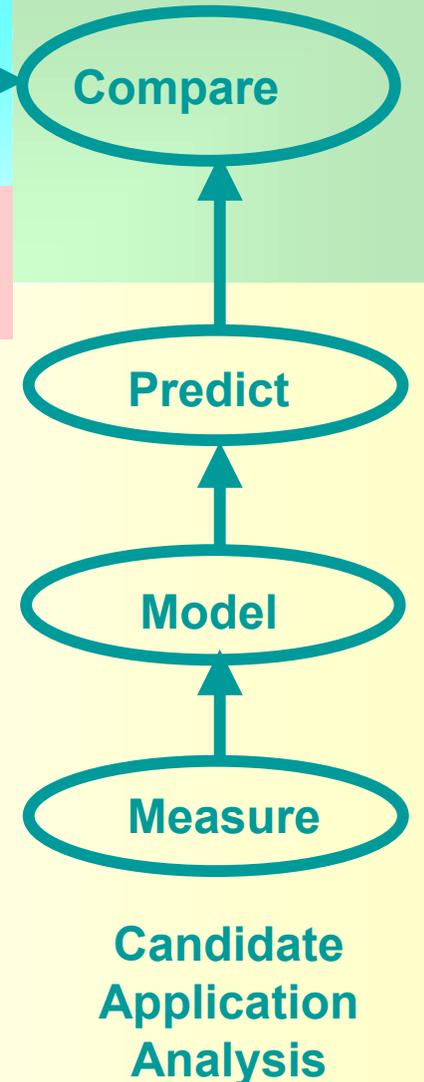
Set performance goals for optoelectronic interconnects

- Quantify performance improvements
- Analyze the factors that affect performance
- Identify engineering challenges
- Define architectural niche for optical interconnect technology

Optical
vs.
Electrical

Identify and characterize the applications that will most benefit from this technology

- Quantify the expected performance improvement
- Identify any new applications that will be enabled





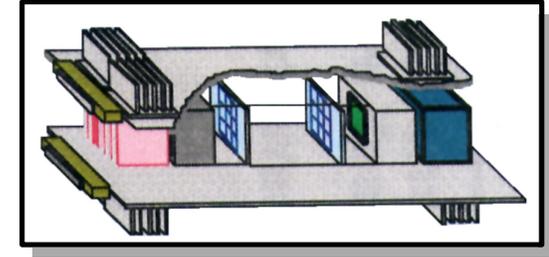
Quantifying Optoelectronic Interconnect Performance



Electronics-Based Signal Processor

Measurements and Analyses

- Bisection bandwidth: balance and scalability with processing power
- Link latency and bandwidth
- Volume and electrical power
- Balance among link, memory and I/O speeds
- Software overhead



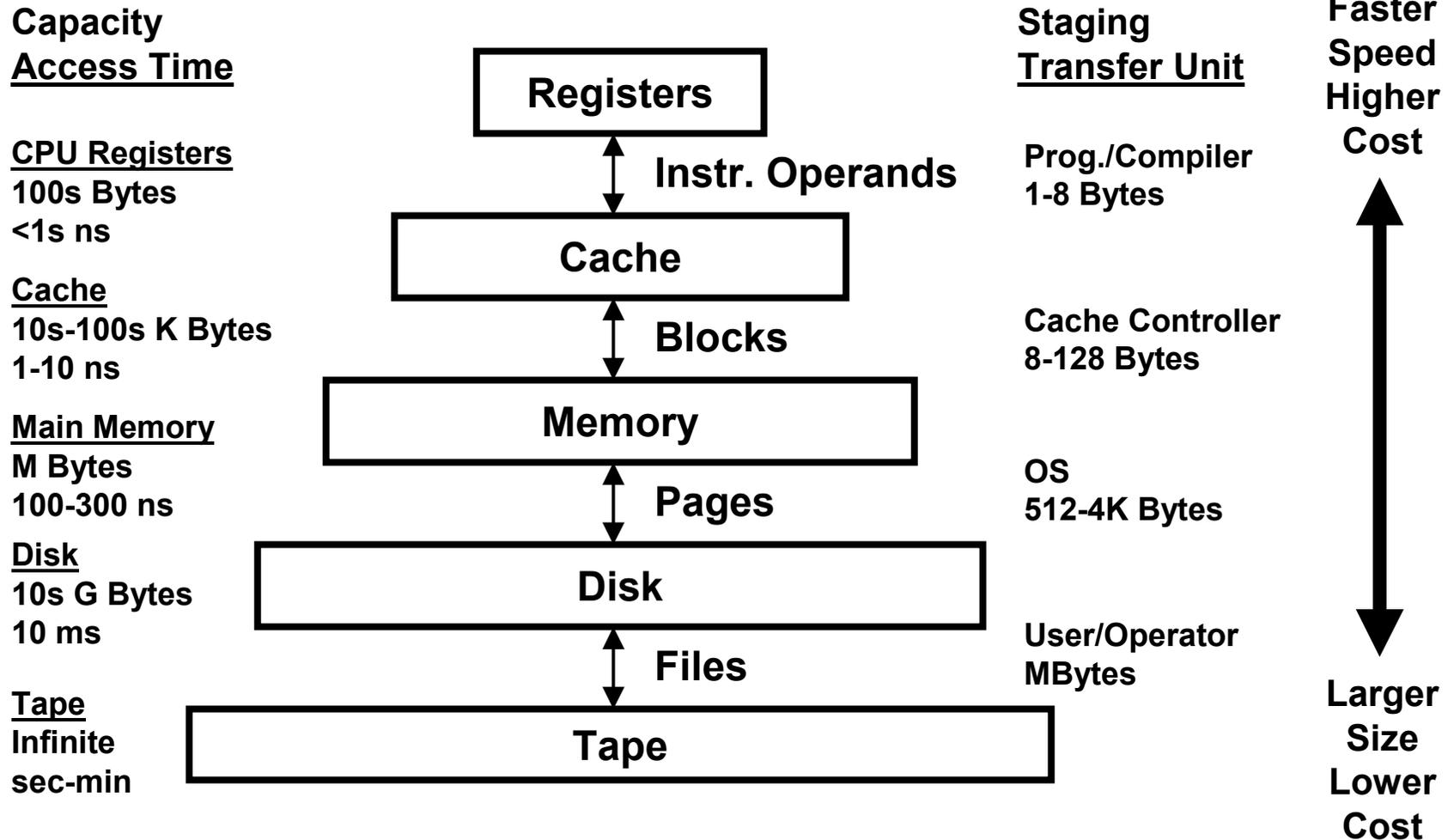
Optoelectronics-Based Signal Processor

Quantifiable Improvements

- Improved bisection bandwidth while meeting volume and electrical power constraints of embedded processors
- Reduced communication overhead
- Increased degrees of parallelism for several important signal processing kernels



Complex Memory Hierarchy



Source: Dave Patterson, Graduate Computer Architecture Course, University of California, Berkeley, Spring, 2001



Outline

- Introduction
- Performance Modeling
- Simulation Roadmap
- Detailed Design
- Summary

- 
- *Program Roadmap*
 - *Benchmarking Summary*



Summary of Benchmarking Results

Pallas MPI SendReceive*

Linear Timing Model

	Startup Latency (μsec)		Bandwidth (MB/s)	
	Measured	Peak	Measured	Peak
Cray T3E-900	20	4	275	960
SGI Origin 2000	53	1	80	160
HP Exemplar	18	-	125	-
Linux Ethernet	220	-	4	12.5
Linux Myrinet	50	1	50	160
Mercury RACE(++)	15	7(4)	145	160(267)
<i>Sky Channel</i>	-	1	270	320
<i>CSPI Myrinet</i>	42	1	157	320

- Optical bandwidth increase improves latency
- Decrease in overhead still needed to improve message time

*<http://www.pallas.com>

LogP-MPI**

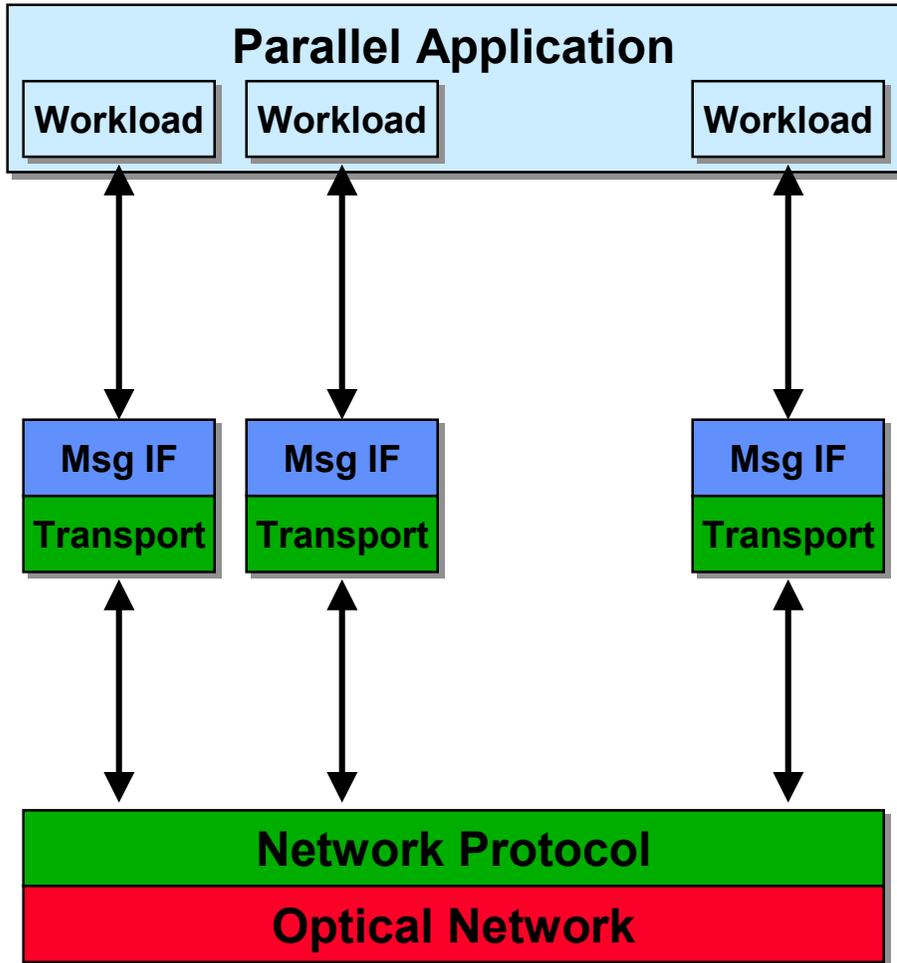
LogP Model

	Parameters (μsec)			Bandwidth		Optical	
	\underline{o}_s	\underline{o}_r	\underline{g}	$\underline{m/g}$ (MB/s)	\underline{L}	$\underline{RTT}_{\text{opt}}$	$\underline{\text{Improvement}}$
Cray T3E-900	8.4	30	7.1	333	11 μs	43 μs	12%
Mercury RACE++	15.5	22.3	21.2	234	1.3 μs	38 μs	3%
SPARC Solaris	353	382.4	521.6	1.1	.5ms	.74ms	41%
Beowulf	12.7	240.4	1998.1	5	.7ms	.26ms	73%
CSPI	46.9	9.5	44.1	110	-	-	-

**<http://www.cs.vu.nl/albatross>



Network Simulation



Measurements

- Computation time
- Communication time (software overhead)

- Bandwidth
- Latency (propagation delay)
- Number of channels

- Overhead (protocol, flow control)
- Average queueing delay (buffer size, arbitration)

- Bandwidth
- Latency (propagation delay)
- Number of channels

- Link Bandwidth
- Latency (protocol, propagation delay)

Example Applications

- HSI
- GMTI
- SAR

Example Workloads

- Corner Turn
- Multicast
- Point to point



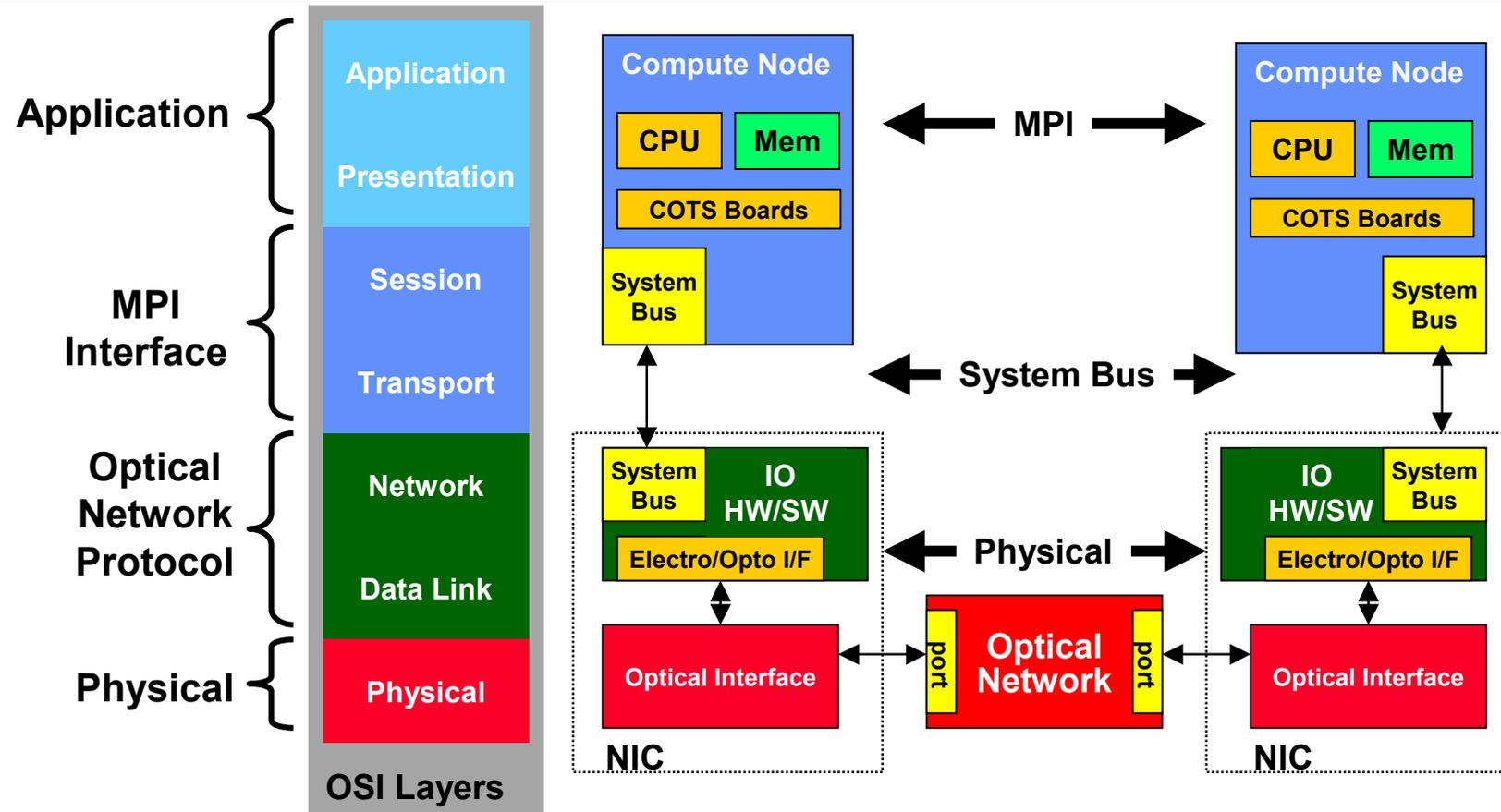
Outline

- Introduction
- Performance Modeling
- **Simulation Roadmap**
- Detailed Design
- Summary

-
- *Simulation Levels*
• *Technical Roadmap*



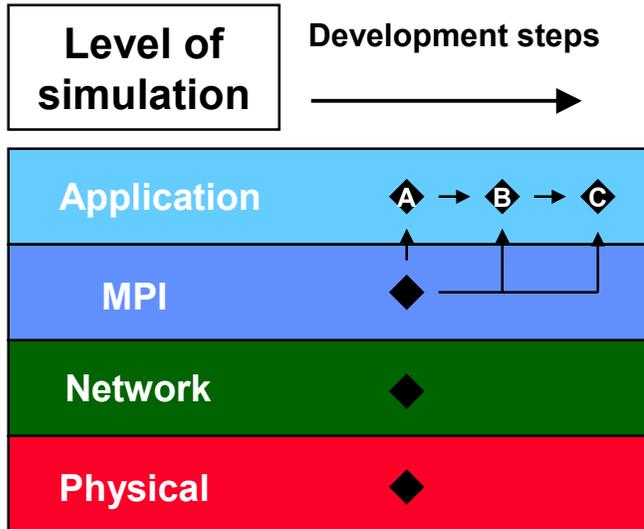
Photonic System Overview



- Simple system consists of two hosts with NICs connected through the network
- The network protocols are mapped to the Open System Interconnect (OSI) layers



Simulation Development Roadmap



Simulation Milestones

FY

A: Point-to-point
- Send-Receive

4Q/01

B: Aggregate
- Corner turn, multi-cast, ...

2Q/02

C: Application
- HSI, GMTI, ...

4Q/02

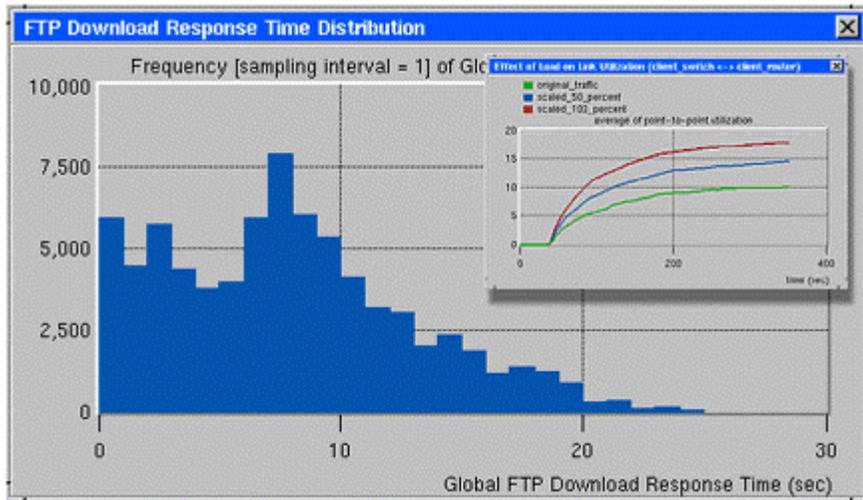
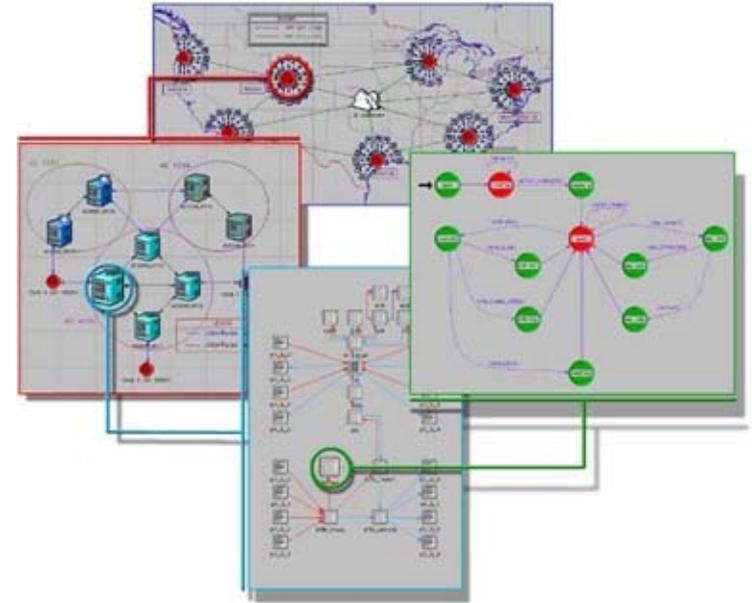
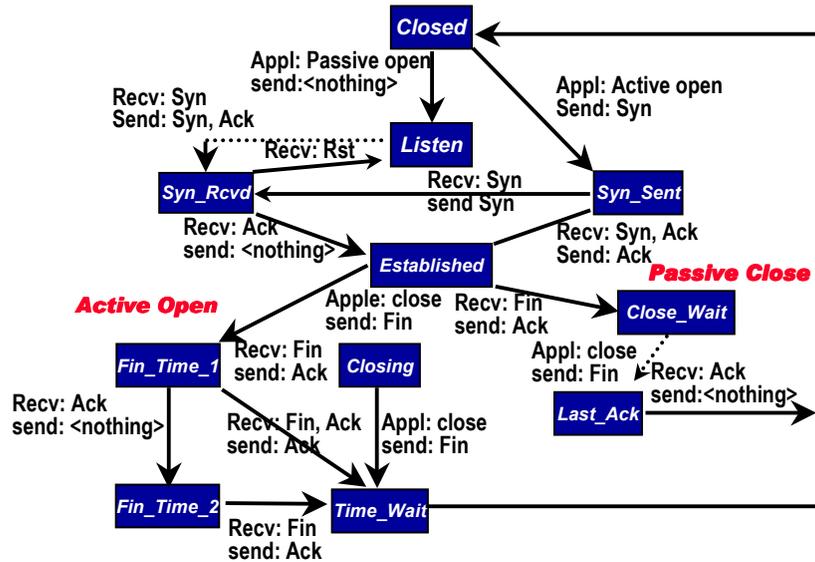
Benefits

- Detailed hardware structural model replaced with general, multi-layer, parameterized model
- Use parameters and topology to contrast electrical and optical interconnects

- Less detailed design information required (parameter estimates only)
- More general applicability to other architectures
- Leverages previous benchmarking activity (parameter derivation)



Opnet Modeler



- Hierarchical network models
- Object-oriented modeling
- Clear and simple modeling paradigm
- Finite state machine modeling
- Comprehensive support for protocol programming
- Wireless, point-to-point and multi-point links



Outline

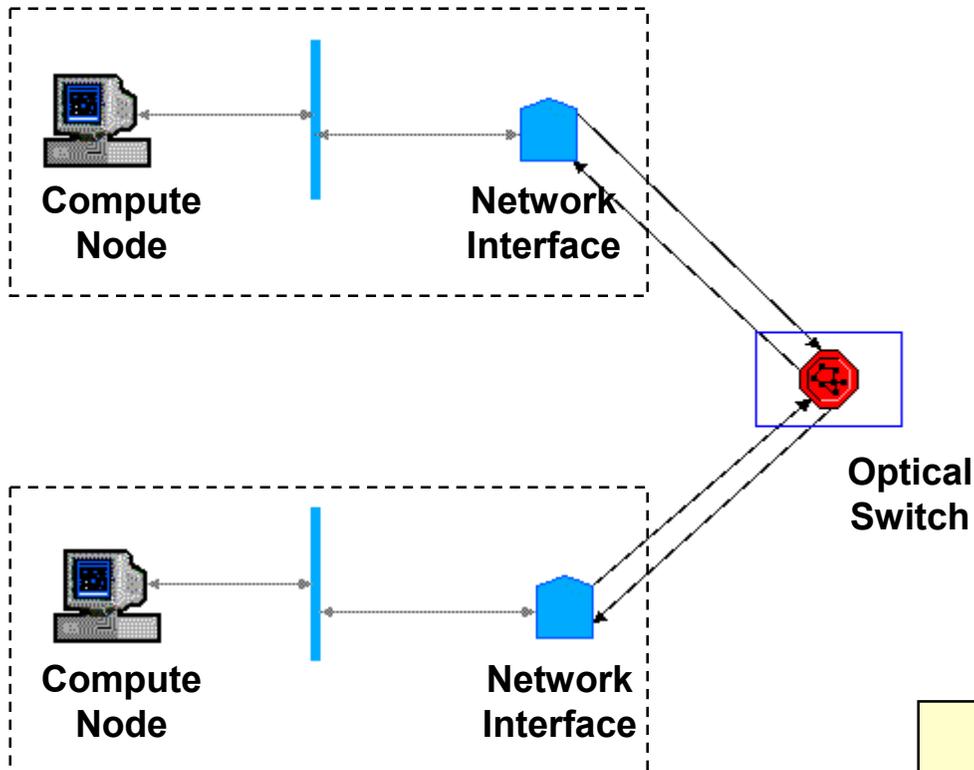
- Introduction
- Performance Modeling
- Simulation Roadmap
- Detailed Design
- Summary

- 
- *Network Components*
 - *Application Simulation*



Simulation: Optical Network

Optical Network Model (two node configuration)



Compute Node

- Application, MPI
- Software Overhead
- Message Protocol

Network Interface

- Data Link Layer
- Protocol Overhead

Optical Switch

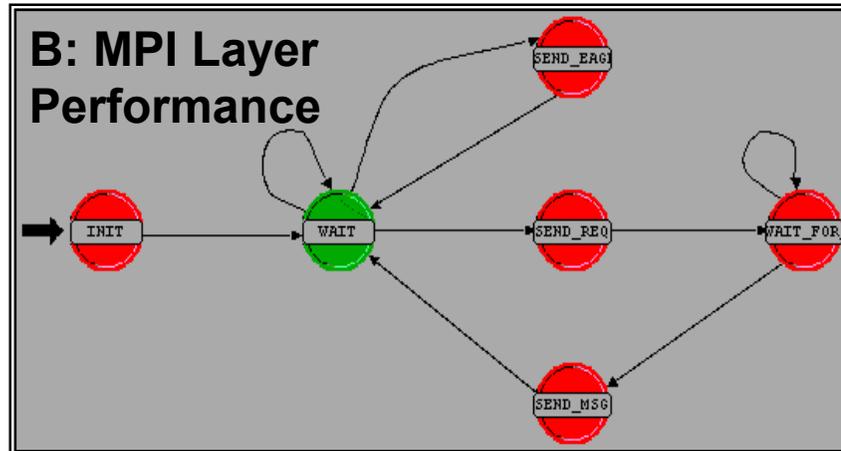
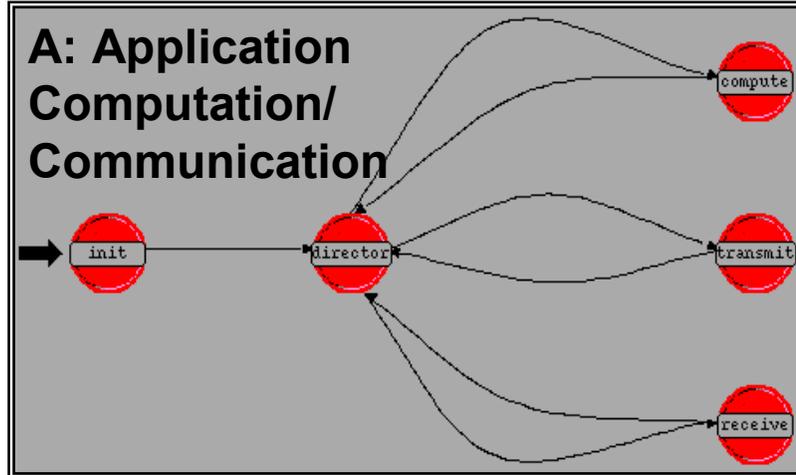
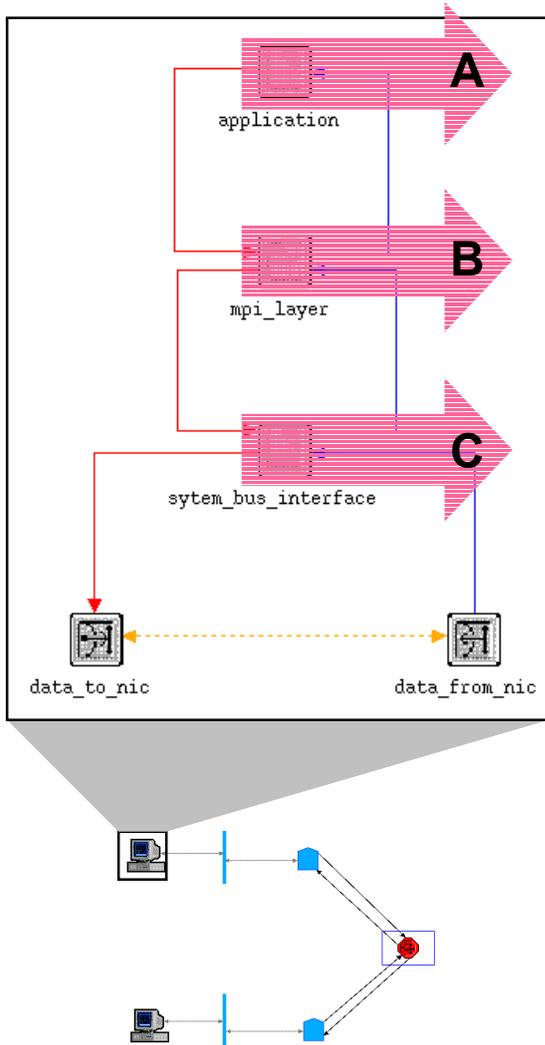
- Fully Connected Topology
- Link/Bisection Bandwidth

**Separate Node models
provide modular functionality**

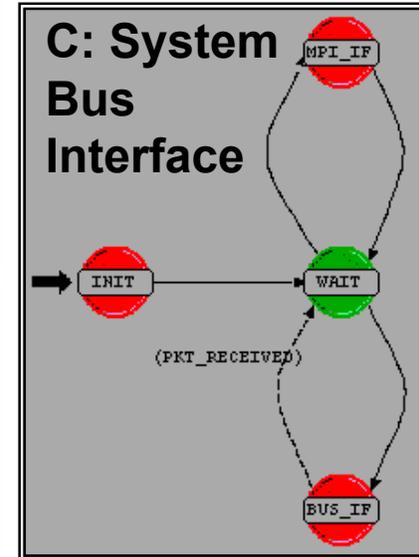


Simulation: Compute Node

Compute Node Model



Process Models



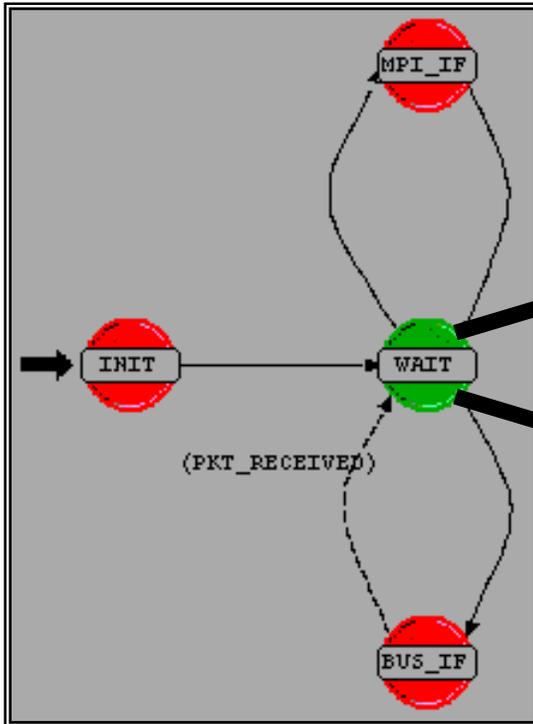
- Future:**
- Infiniband
 - RapidIO

State diagrams implement protocols at each level



Simulation: Sample State Code

System Bus Interface Process Model



C-code implements functionality and gathers statistics

System bus wait entry code

```

/* get the intrpt stream index
stream_index = op_intrpt_strm

/* get a pointer to the packet
pkptr = op_pk_get(stream_index)

#ifdef PCI_DEBUG1
printf("\npci_interface process\n");
op_pk_print(pkptr);
#endif

/* determine whether the packet is
PCI bus receiver stream or the
if (stream_index == STRM_FROM_UPPER_LAYER)
{
/* if the packet came from the
* write the statistics for the
* encapsulate the packet
* send the encapsulated packet
*/
}
else
{
/* else the packet came from the upper layer
* (stream_index == STRM_FROM_UPPER_LAYER) so do this...
*/
}

```

Statistics Options:

- Global (all processes)
- Local (one process)

Statistics Type:

- Counter (index)
- Delay (sec)
- Data Size (bits)
- Data Rate (bits/sec)

```

/* Calculate metrics to be updated.
pk_size = op_pk_total_size_get(pkptr);

#ifdef PCI_DEBUG1
/* Update local statistics.
*/
op_stat_write(bits_rcvd_stathandle, (double)pk_size);
op_stat_write(pkts_rcvd_stathandle, 1.0);

op_stat_write(bitssec_rcvd_stathandle, (double)pk_size);
op_stat_write(bitssec_rcvd_stathandle, 0.0);
op_stat_write(pktssec_rcvd_stathandle, 1.0);
op_stat_write(pktssec_rcvd_stathandle, 0.0);
#endif

/* create an encapsulation packet */
encap_pkptr = op_pk_create_fmt("tbe_pkt2");

/* encapsulate the original packet in the encapsulation packet */
op_pk_nfd_set(encap_pkptr, "encap_tbe_pkt", pkptr);
op_pk_nfd_set(encap_pkptr, "source", 999);
op_pk_nfd_set(encap_pkptr, "destination", 555);

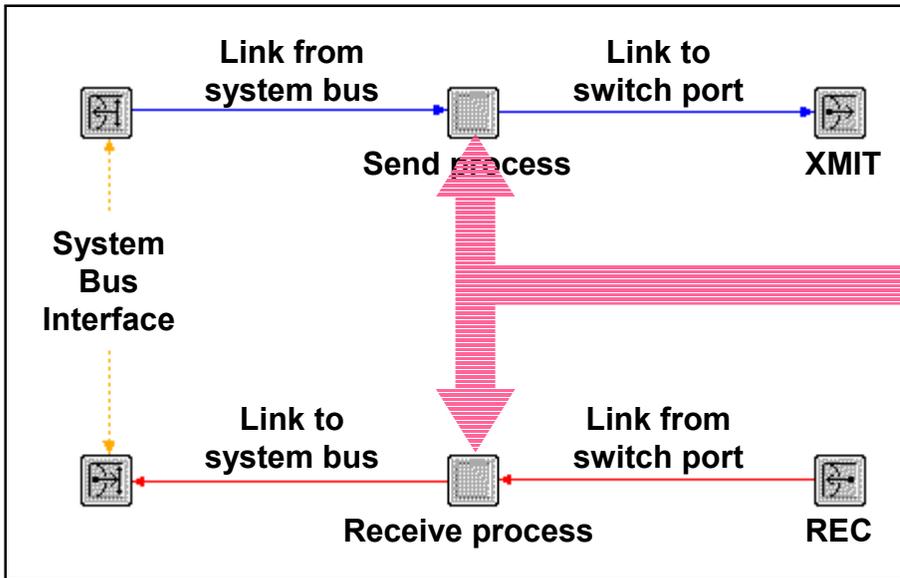
/* send the encapsulation packet */
op_pk_send(encap_pkptr, STRM_TO_UPPER_LAYER);
}
else
{
/* else the packet came from the upper layer
* (stream_index == STRM_FROM_UPPER_LAYER) so do this...
*/
}
}

```

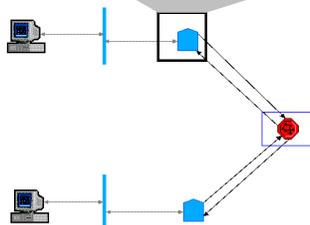
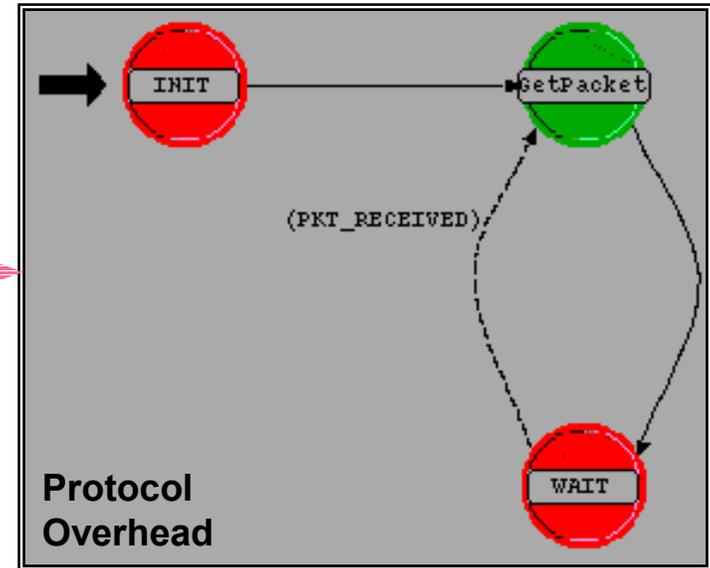


Simulation: Network Interface

Network Interface Node Model



Send and Receive Process Models

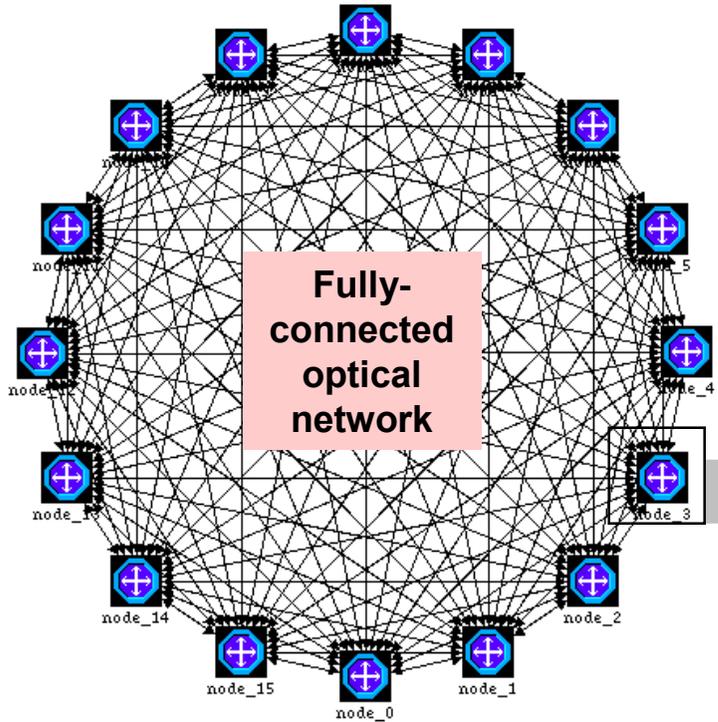


State diagrams implement message queues, flow control and transfer protocols



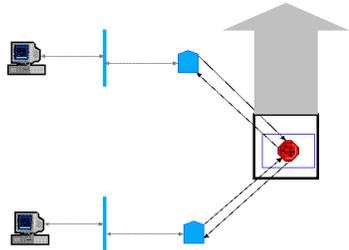
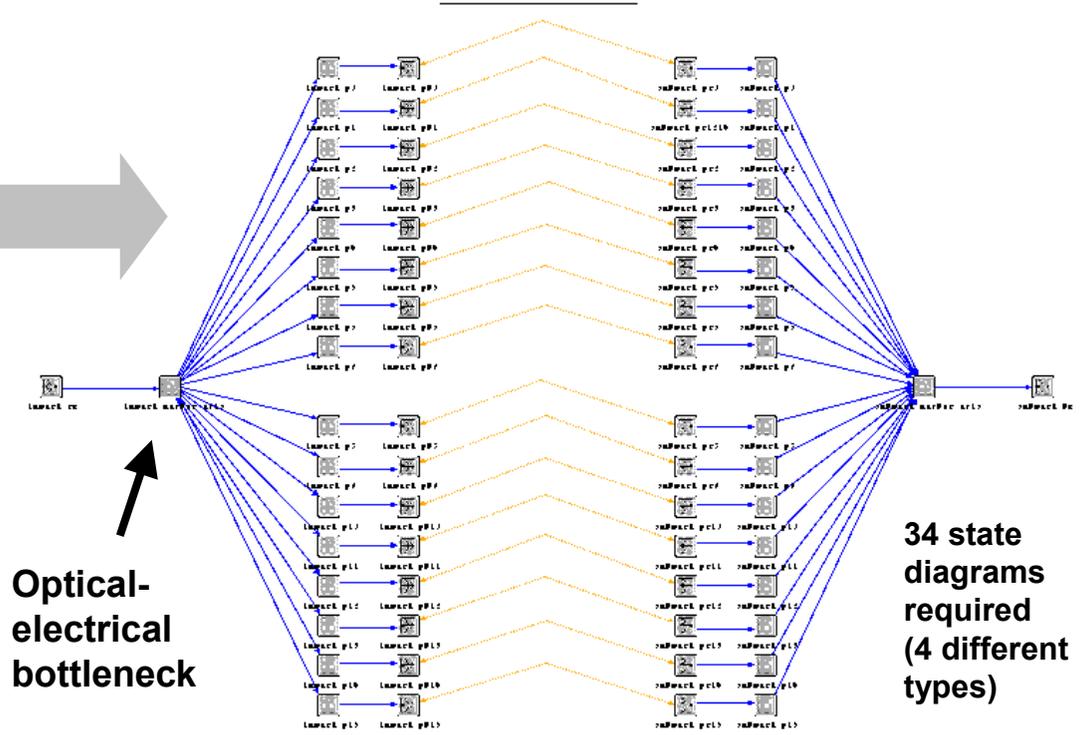
Simulation: Optical Switch

Optical Switch Network Model



Simulate full optical scalability vs. electrical interface capability

Switch Port Node Model

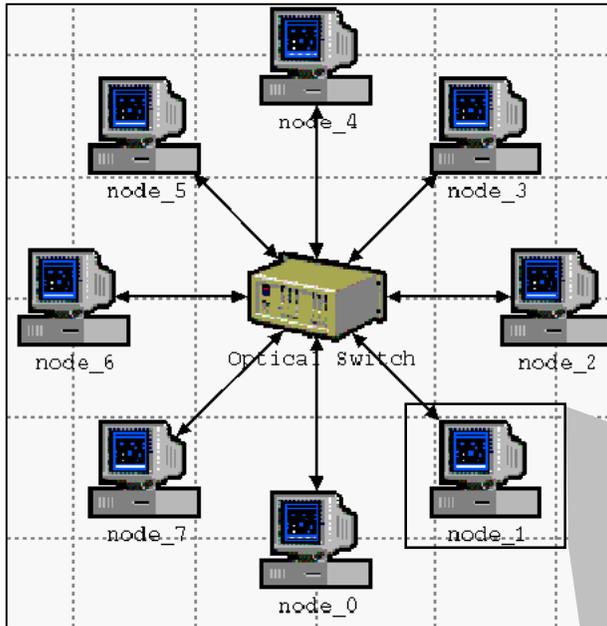


Optical-electrical bottleneck

34 state diagrams required (4 different types)



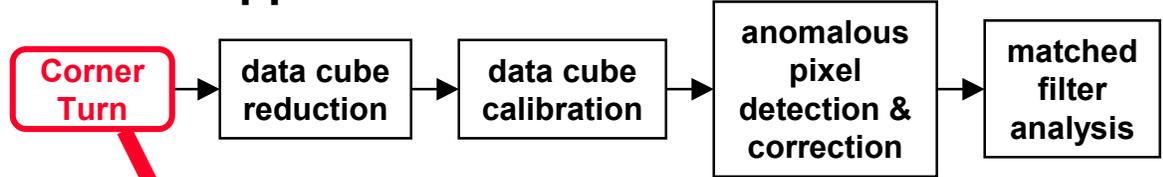
Application Modeling



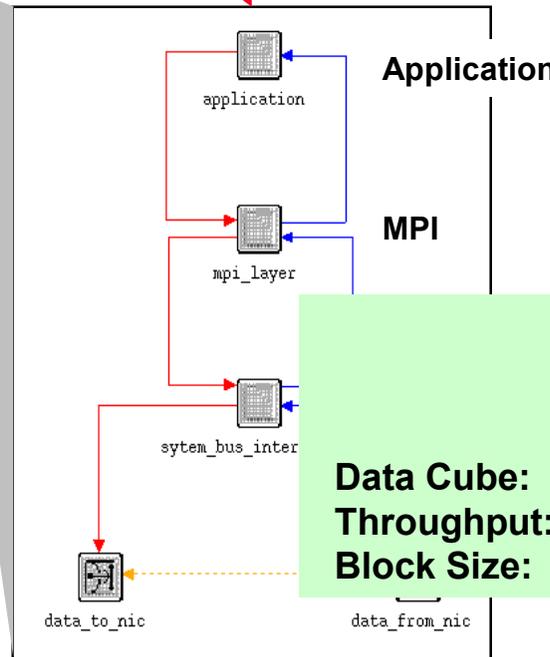
Eight-node Optical Network

- Performance metrics gathered at each layer

HSI Application



Application Workload Analysis: Corner-Turn



	Test-Bed	Today	2005
Data Cube:	186 MB	2.9 GB	188 GB
Throughput:	186 MB/s	2.9 GB/s	188 GB/s
Block Size:	23 MB	360 MB	24 GB



Summary

- **Simulation strategy focuses on system-level analysis vs. detailed component level**
 - Less detailed design information required (parameter estimates only)
 - More general applicability to other architectures
 - Leverages previous benchmarking activity (parameter derivation)
- **Simulation focuses on system-level optical network performance**
 - Prediction for representative workloads
 - Requirements derived from current and future applications