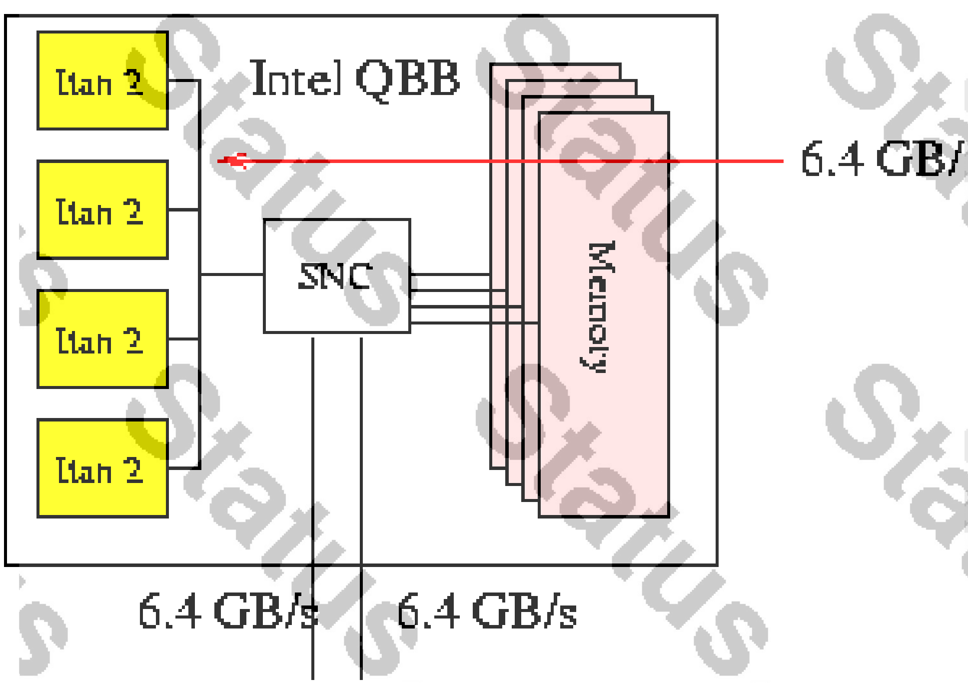
Towards Ubiquitous Supercomputing

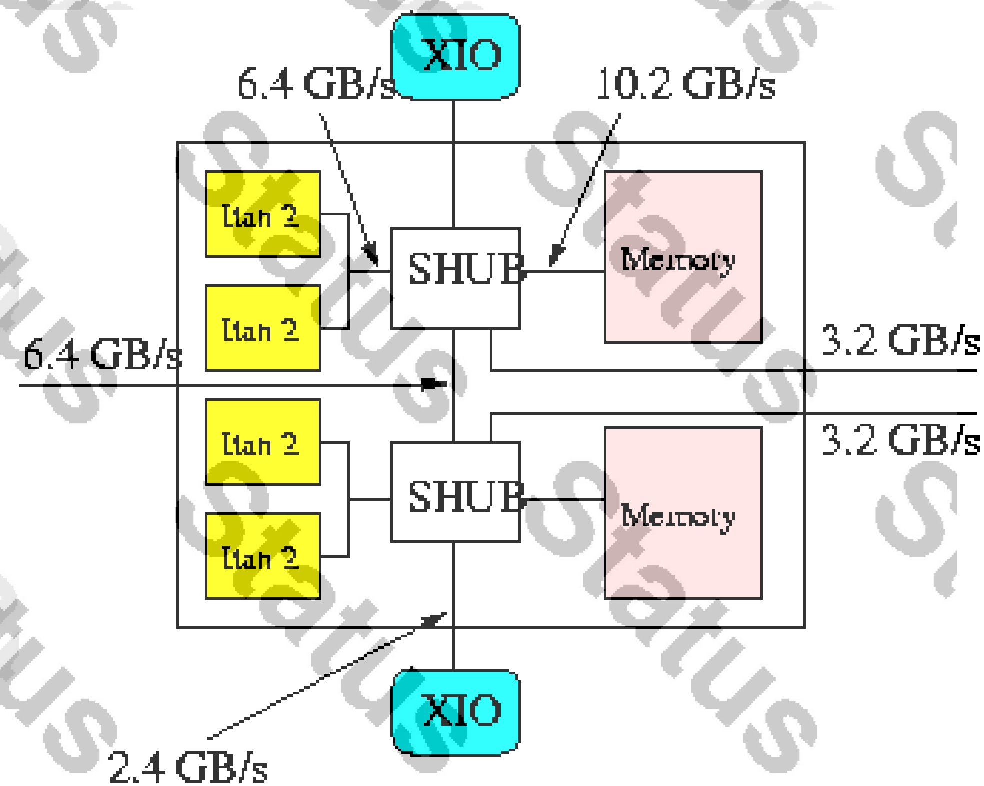
Like the universe and our perception of it are shaped by the speed of light and gravity, so is High Performance Computing (HPC) and our perception of it shaped by latency and bandwidth.

- Where do we stand?
- How to gain ground in the latency/bandwidth battle?
- How does it effect our computations?

The Status 1

HPC systems presently consist largely of a collection of identical standard processors connected by an internal network. We have bandwidth and latency constraints within and between processors:





The Status 2

Bandwidths are converging:

- Intranode: $\simeq$ 10 GB/s.
- Internode: $\simeq$ 1 GB/s.
- Intersystem: $\simeq$ 1GB/s.

But ...

The Status 3

Latencies differ vastly:

— Intranode: $\simeq$ 100 ns (10^{-9} s).

— Internode: $\simeq$ 2 – 10 μs (10^{-6} s).

— Intersystem: 3.3ns/m (So, in a grid we talk about ms, 10^{-3} s).

In intranode and internode latencies we still have a (small) world to win.

The Status 4

Parameters for present-day clusters:

	Bandwidth MB/s	Latency µs
Gbit Ethernet	120	≥ 40
Infiniband	850	7
Myrinet II	250	10
QsNet	400	4
$QsNet^{II}$	980[†]	2
SCI	500	2

[†]Constrained by PCI-X bus, > 1000 MB/s

What is done?

What is done presently: increase about everything on chip.

Development	Effect	Net result
— Clock frequency	Widens Memory-CPU gap	–
— Cache size	Tends to be slower	–
— Process threads	Can take advantage of stalling memory requests	+
— # of processor cores	Shares memory bandwidth	–
— Network on chip	Topology becomes important	+/–

What should be done?

Most present developments are not addressing HPC needs: price/performance *not* absolute performance is driving them. So:

- Increase speed of memory.
- Decrease memory latency.
- Increase memory bandwidth.
- Diversify compute engines.
- Turn around computer architecture.

What about memory speed?

— Present development: FC-RAM (Fast Cycle RAM)

About 3 times faster than fastest DDR2-SDRAM.

— Near future: MRAM (Magnetic RAM)

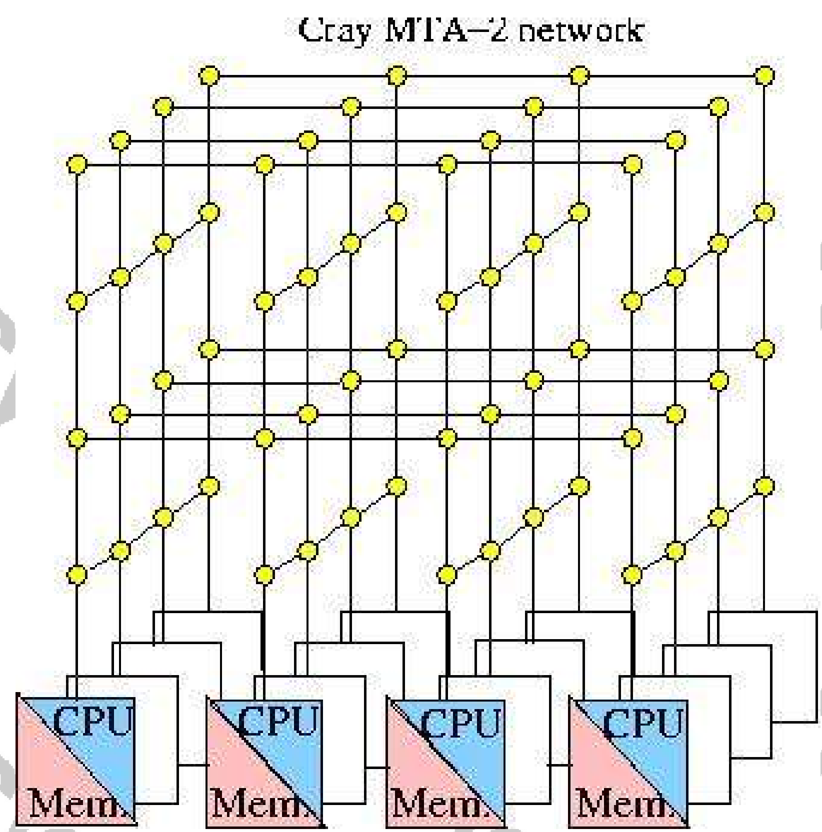
- As fast as FC-RAM (to begin with).
- Permanent, like SRAM.
- There is still a density problem.

What about memory latency?

- Forego it by pipelining memory requests and direct access to registers: vector processors.

- Hide it by latency tolerant architecture: multithreading. Cray MTA-2:
  - 128 threads/processor.
  - Switch between threads 1 clock cycle.

- Never leave the memory.



Processing in Memory 1

Processing in Memory (PIM), also called Computation in RAM (C-RAM).

Idea: Why ship operands to/from memory from/to CPU when processing can be done in memory?

So: enhance memory cells slightly with bit-wise processor to do massive SIMD type computing.

- Was part of HTMT Petaflop project (defunct).
- IBM, Cray still interested and active.
- Reminiscent to 1980s processor-array systems ICL DAP, Goodyear MPP, ...
- Helpful for massive, regular data processing.

Processing in Memory 2

Speedups of PIM vs. host computer:

16M 3×3 Convolution	: 6404
Vector quantisation	: 1312
Data Mining	: 2724

D.G. Elliott, W.M. Snelgrove, M. Stumm, *A PetaOp/s is currently feasible by computing in RAM*, PetaFlops Frontier Workshop, Washington, 1995.

How to increase memory bandwidth? 1

Increasing the memory bandwidth only makes sense when:

1 — The memory is fast enough to satisfy the request.

2 — The frontside bus of the processor is able to accept the data.

With standard processors the frequency/width of the frontside bus mostly determines the bandwidth allowed. So:

Use non-standard processors.

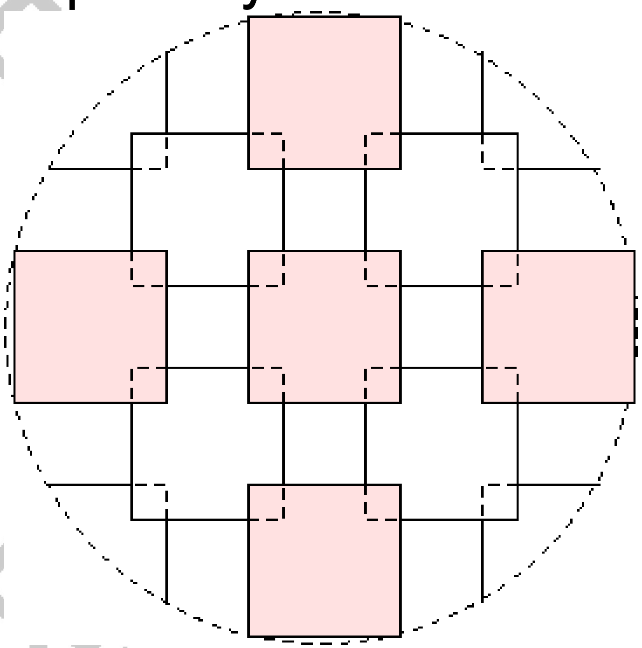
How to increase memory bandwidth? 2

Non-standard processors are available and with increasing functionality and speed: FPGAs (Field Programmable Gate Arrays).

- Device density 10 times higher than that of standard processor.
- Can be configured to implement
  - Memory and I/O interfaces.
  - Many important algorithms (speedups of orders of magnitude).
- Begin to be adopted by HPC vendors.

How to increase memory bandwidth? 3

There are also proprietary solutions like pursued by SUN: Chips are "connected" by stacking them with extremely close proximity of devices on neighbouring chips. The result is that the currents become coupled by capacitance: proximity bonding.



- High bandwidth (> 20 GB/s).
- Low Latency (50 – 100 ps).
- Still figuring out alignment and bonding.

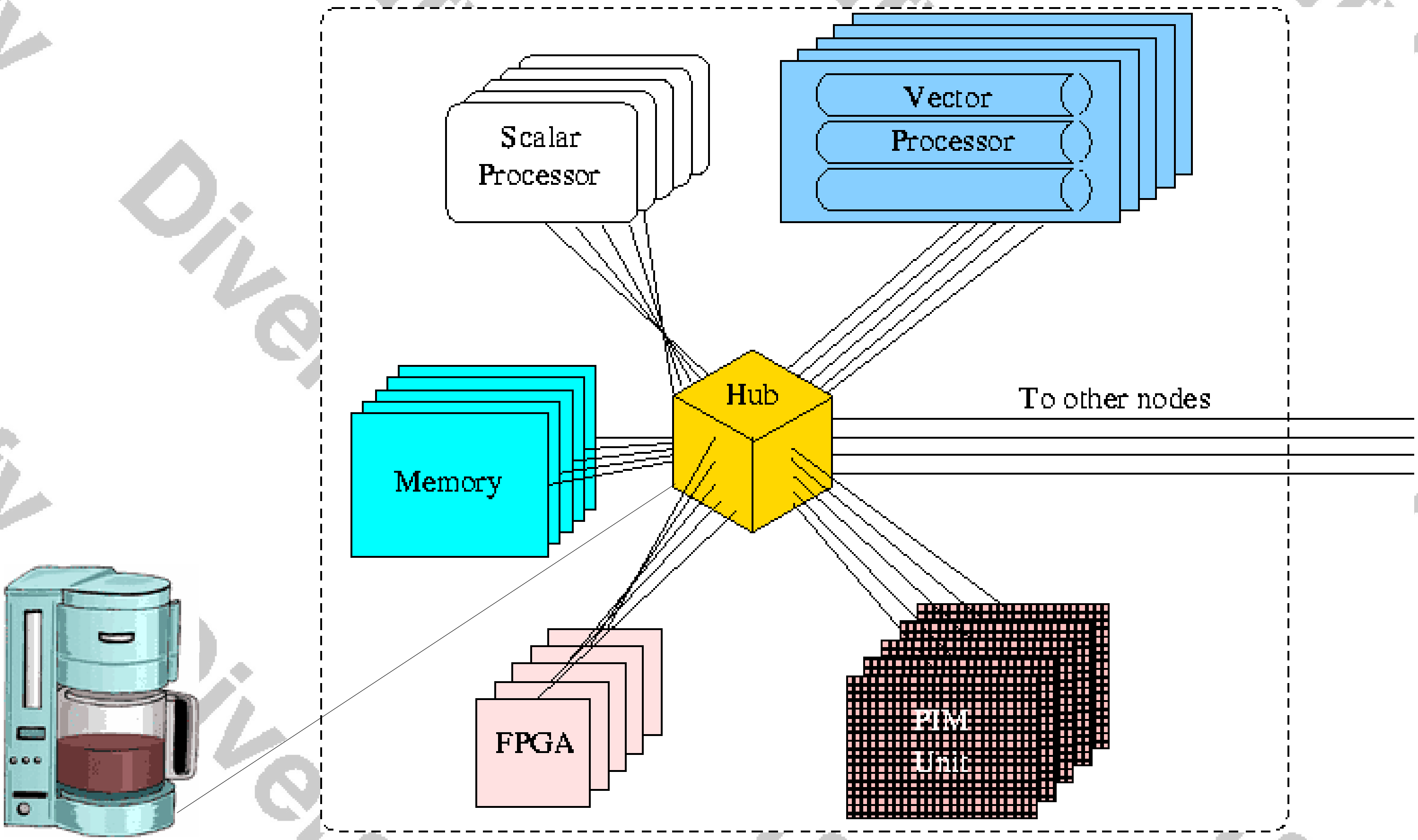
Diversifying the system 1

To take maximal advantage of an HPC system it should diversify in the type of processors and memory:

	MIPS R14000 500 MHz Mflop/s	Itanium 2 1.5 GHz Mflop/s	Cray X1 800 MHz, 1MSP Mflop/s
axpy	328	2707	2579
1st order recursion	244	143	46

Diversifying the system 2

A node of such a system could look like this:



Turning the architecture around 1

The number and type of processors and memory in systems may be different. Even nodes need not be the same.

The constant factor is the system infrastructure:

The network becomes the computer.

Turning the architecture around 2

Turning the architecture inside out can be taken one step further.

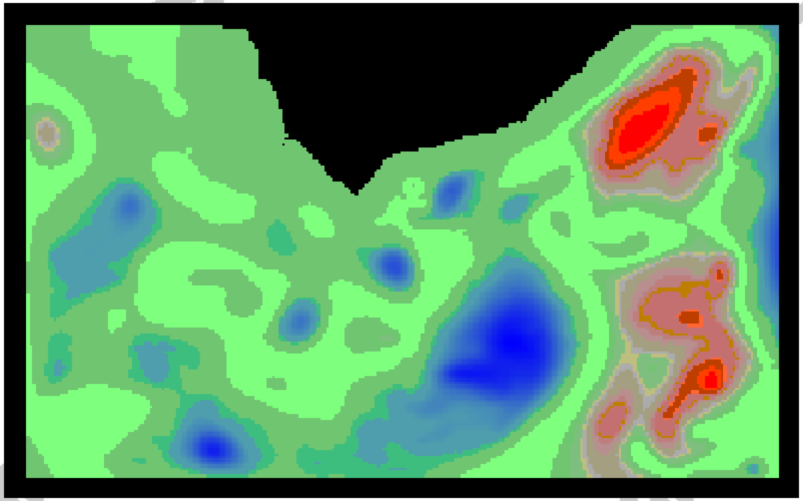
The devices within the systems will become

separately addressable

(IPv6?) and suddenly one has a huge spectrum of diverse compute/storage/IO devices available. We are approaching the state of Ubiquitous Supercomputing.

Distributed Applications 1

Stochastic ensemble simulations are ideal for such distributed systems:



Ocean model of South Africa's Agulhas Bay Area

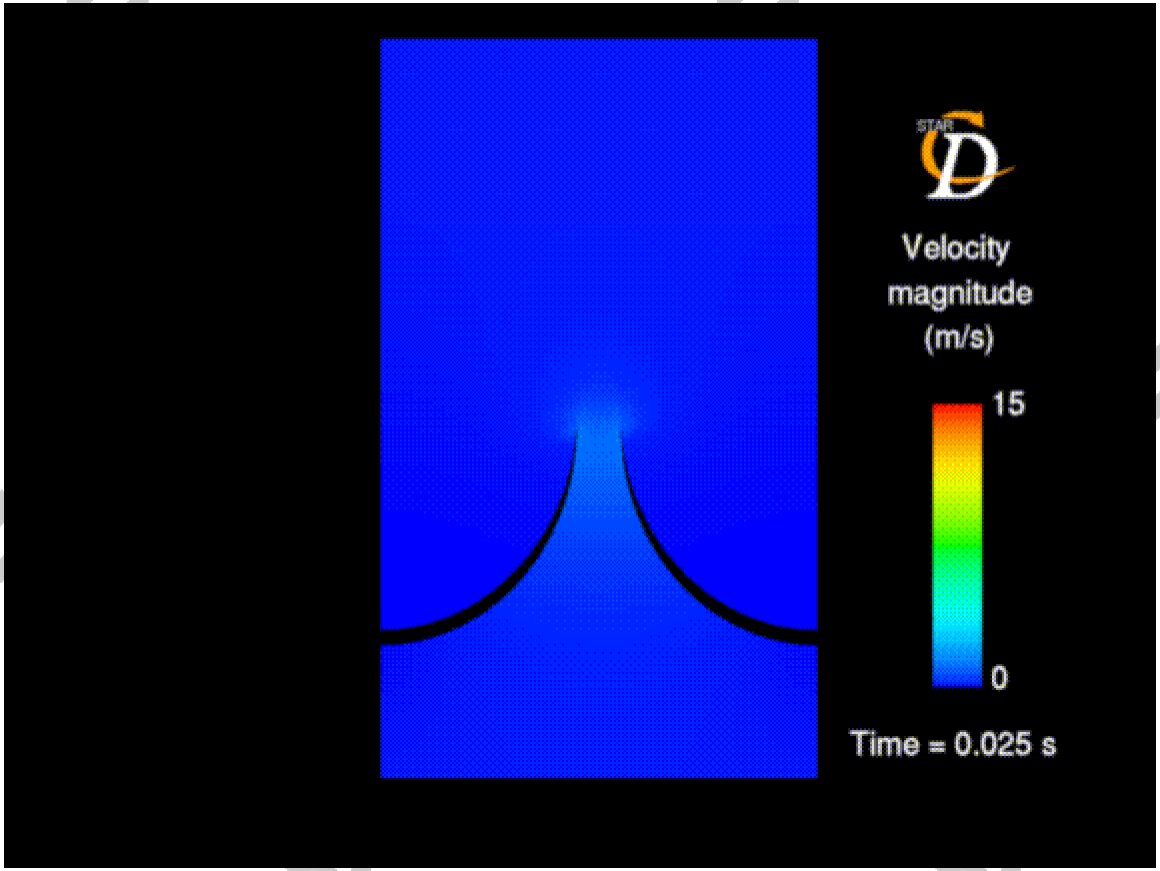
Distributed Applications 2

Sensor networks fused with clusters and very large integrated HPC systems:

FPGAs, clusters, BlueGene/L

Non-Distributed Applications

Do not do this on a geographically distributed system:



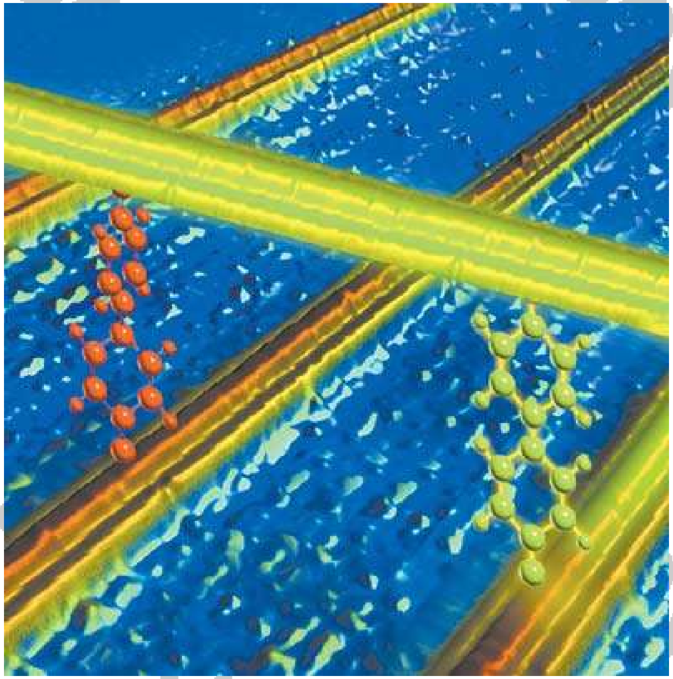
Hart valve model using data redistribution

Beyond Silicon 1

Somewhere in 2012–2014 the feature size of switching elements on Si-based chips will have decreased to 7–8 nm (10^{-9}m).

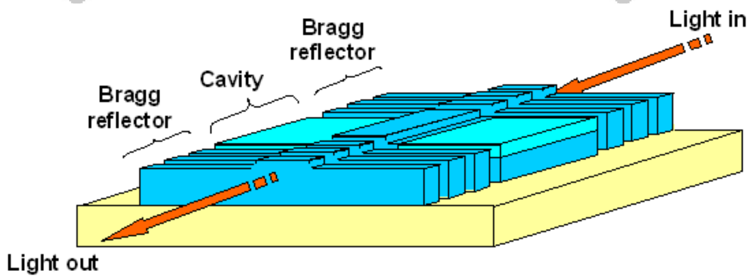
This is a lower bound for switches on a Si substrate.
We have to revert to other (nano)technologies, like:

- HP's crossbar latch (Kuekes, *et.al.*, HP Research).

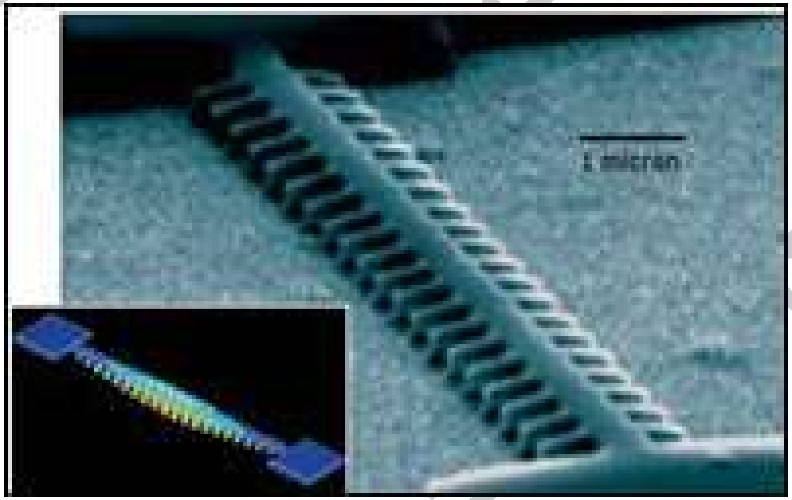


Beyond Silicon 2

- Opto-electronics:
  Optical switches have been realised of 50–200 nm with switching times of about 100 fs (10^{-15}s).
  (Lipson, *et.al.*, Cornell)



- Nano-mechanical quantum devices:
  (Nori *et.al.*, RIKEN/Ann Arbor)



Ubiquitous Supercomputing

Do these new developments alter the scientists' life?

— YES. Ubiquitous Supercomputing lies around the corner with:
  - Vastly increased compute power.
  - Vastly increased latency ranges.
  - Vastly increased modeling possibilities.

— NO. There is always a next problem, a next bottleneck, a next bug to be resolved, and a next solution.

The End