Design of a Software Correlator for the Phase I SKA

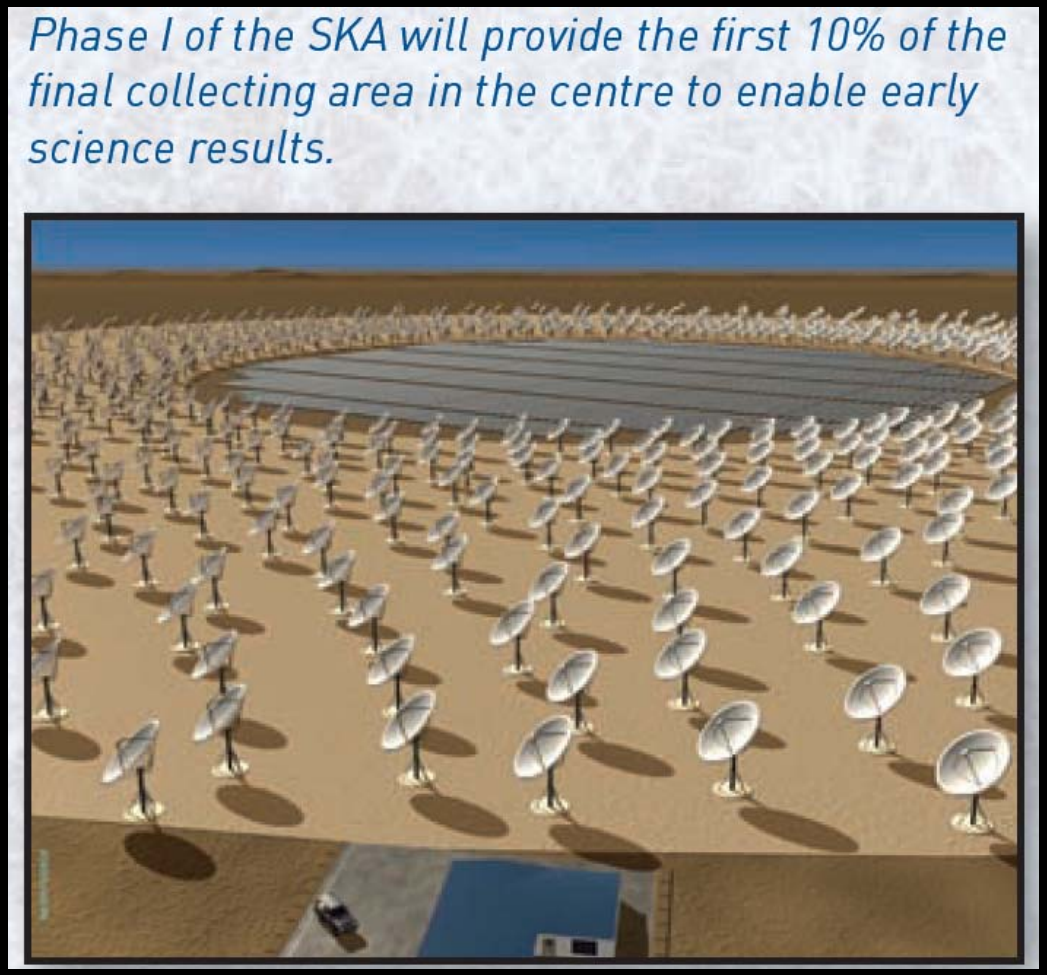
Jongsoo Kim
Cavendish Lab., Univ. of Cambridge &
Korea Astronomy and Space Science Institute

Collaborators: Paul Alexander (Univ. of Cambridge)
Andrew Faulkner (Univ. of Cambridge)
Bong Won Sohn (KASI, Korea)
Minho Choi (KASI, Korea)
Dongsu Ryu (Chungnam Nat'l Univ., Korea)

Correlators for Radio Interferometry

- ASIC (Application-Specific Integrated Circuit)
- FPGR (Field-Programmable Gate Arrays)
- Software (high level-languages, e.g., C/C++)
  - Rapid development
  - Expandability
  - …

Phase I SKA (2015-2018)



- 10% of the final collecting area
- ~300 dishes
- software correlator

Correlation Theorem, FX-correlator

$$R_i(f) = \int_{-\infty}^{+\infty} r_i(t) \mathrm{e}^{2\pi i f t} dt$$

F-step (FT):
~$\log_2(N_c)$ operations per sample

$$\int_{-\infty}^{+\infty} r_i(\tau + t) r_j(\tau) d\tau \Leftrightarrow R_i(f) R_j^*(f)$$

X-step (CMAC):
~ N operations per sample

FLOPS of the X-step in FX correlator

$$4\times 8\frac{N(N+1)}{2}\left(\frac{B}{\mathrm{Hz}}\right)[\mathrm{FLOPS}]\approx 16N^2\left(\frac{B}{\mathrm{GHz}}\right)[\mathrm{GFLOPS}]$$

- 4 is from $R_iR_j^*, R_iL_j^*, L_iR_j^*, L_iL_j^*$
- 8 is from 4 multiplications and 4 additions:

$$+R_iR_j^* = +(a_i+ib_i)(a_j-ib_j) = +(a_ia_j+b_ib_j)+i(b_ia_j-a_ib_j)$$

- $N(N+1)/2$ is the number of auto- and cross-correlations with antenna N
- if B (bandwidth) = 4 GHz

  N=2 → 96x4 GFLOPS = 384 GFLOPS

  N=100 → 16x4x10^4 GFLOPS = 640 TFLOPS

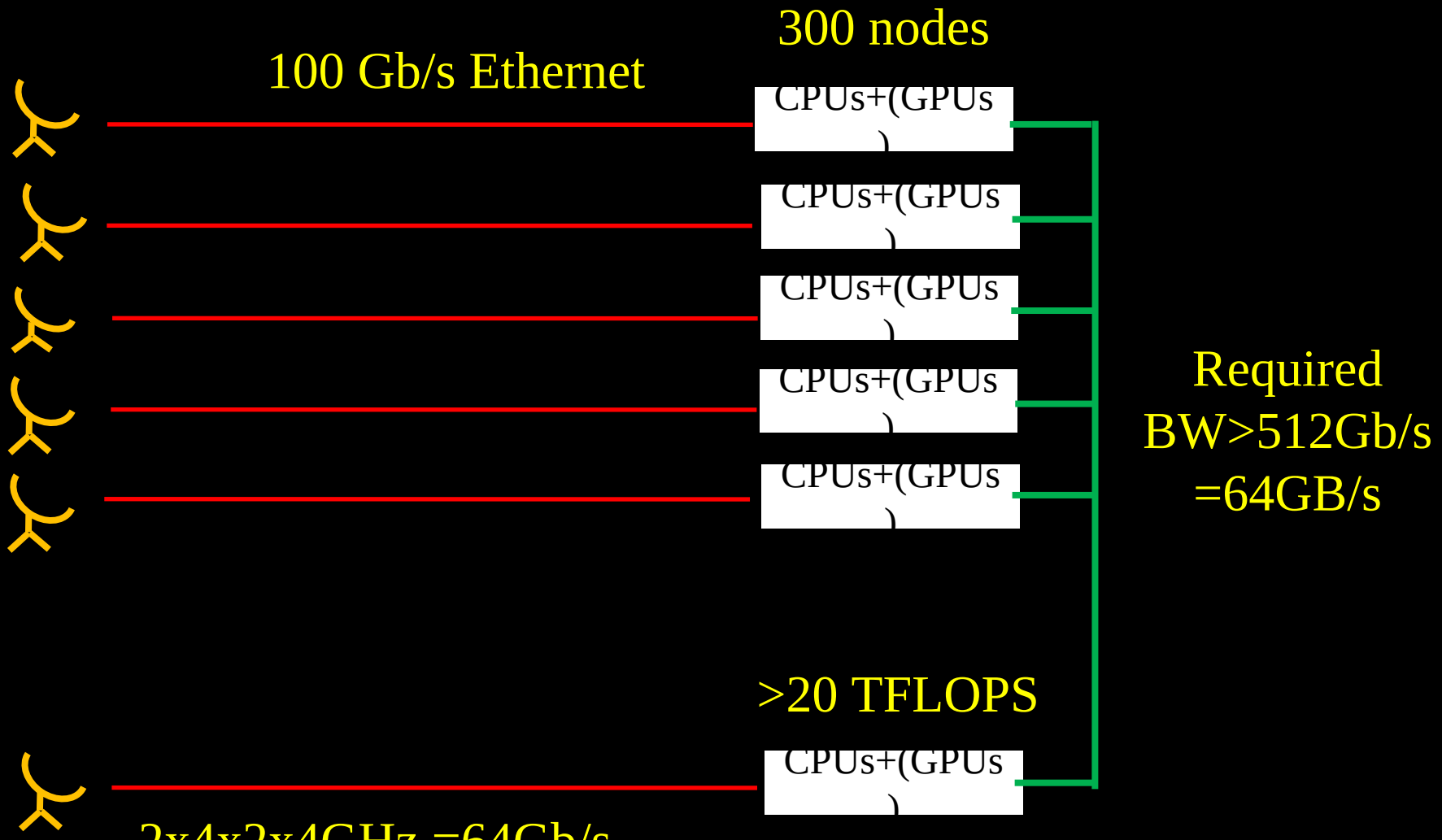
  N=300 → 16x9x4x10^4 GFLOPS = 5.76 PFLOPS

  N=3000 → 16x9x4x10^6 GFLOPS = 576 PFLOPS

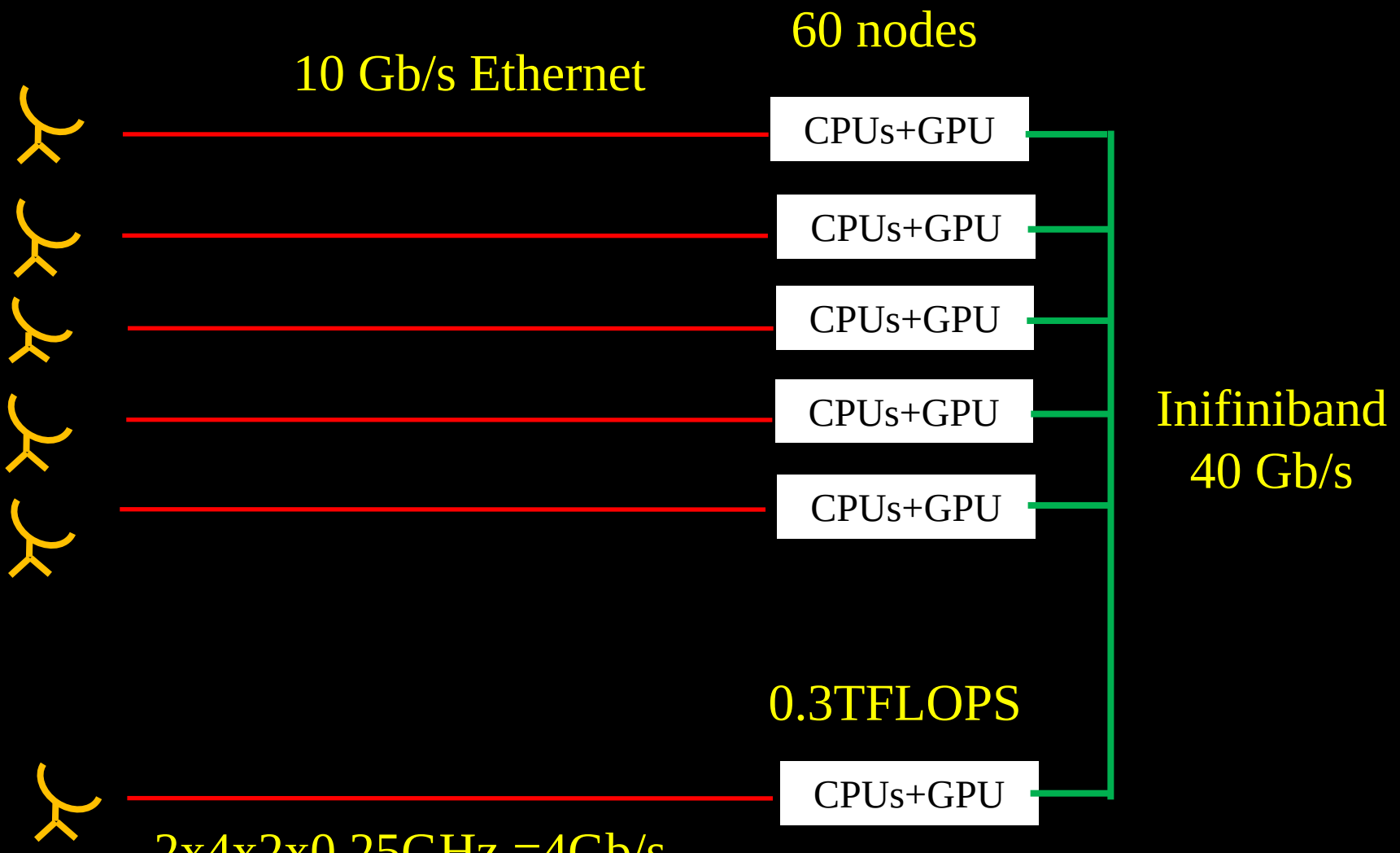
Design goals

- Connect antennas and computer nodes with simple network topology
- Use future technology development of HPC clusters
- Simplify programming

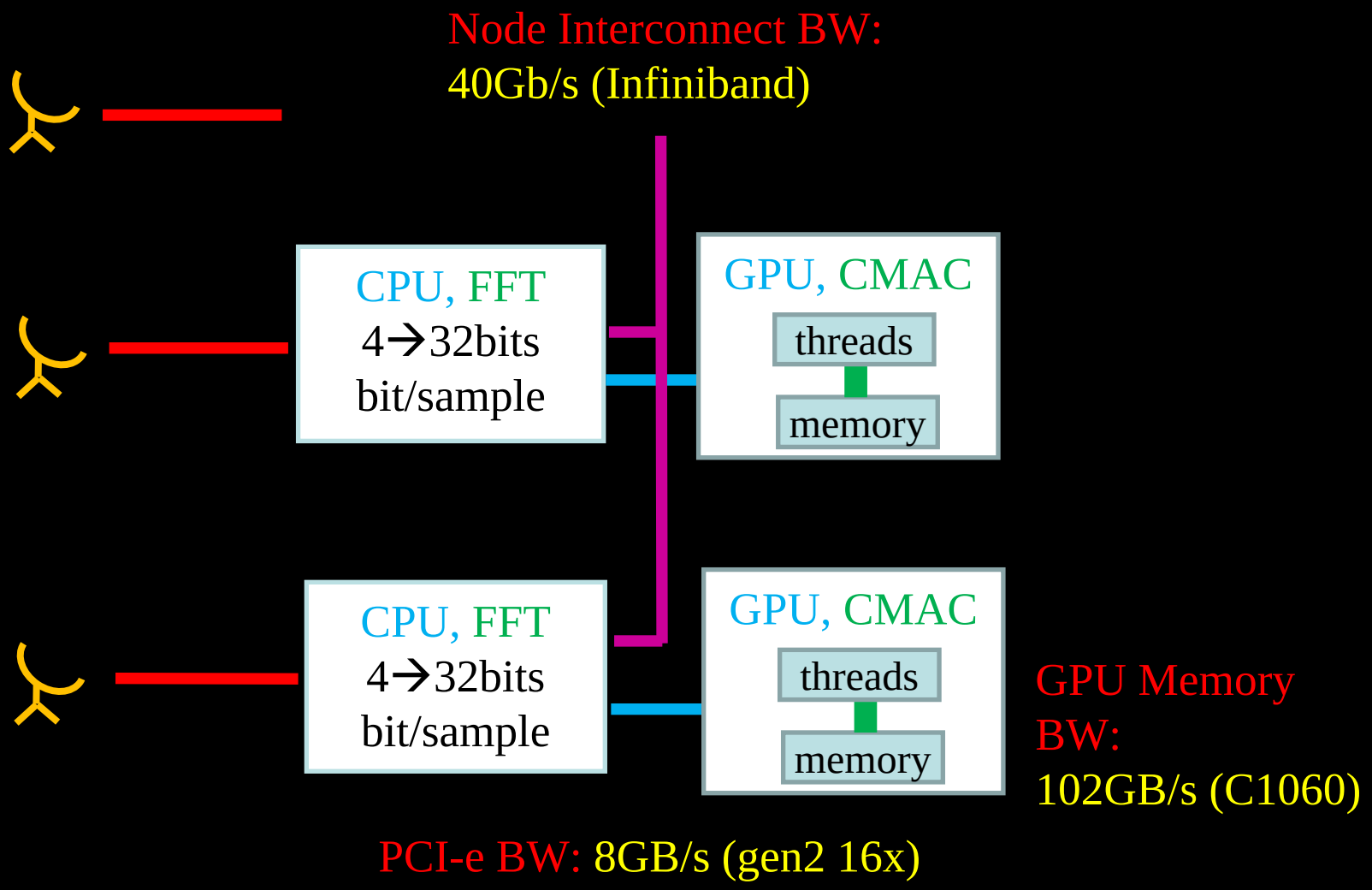
Software Correlator for Phase I SKA (‘2018)



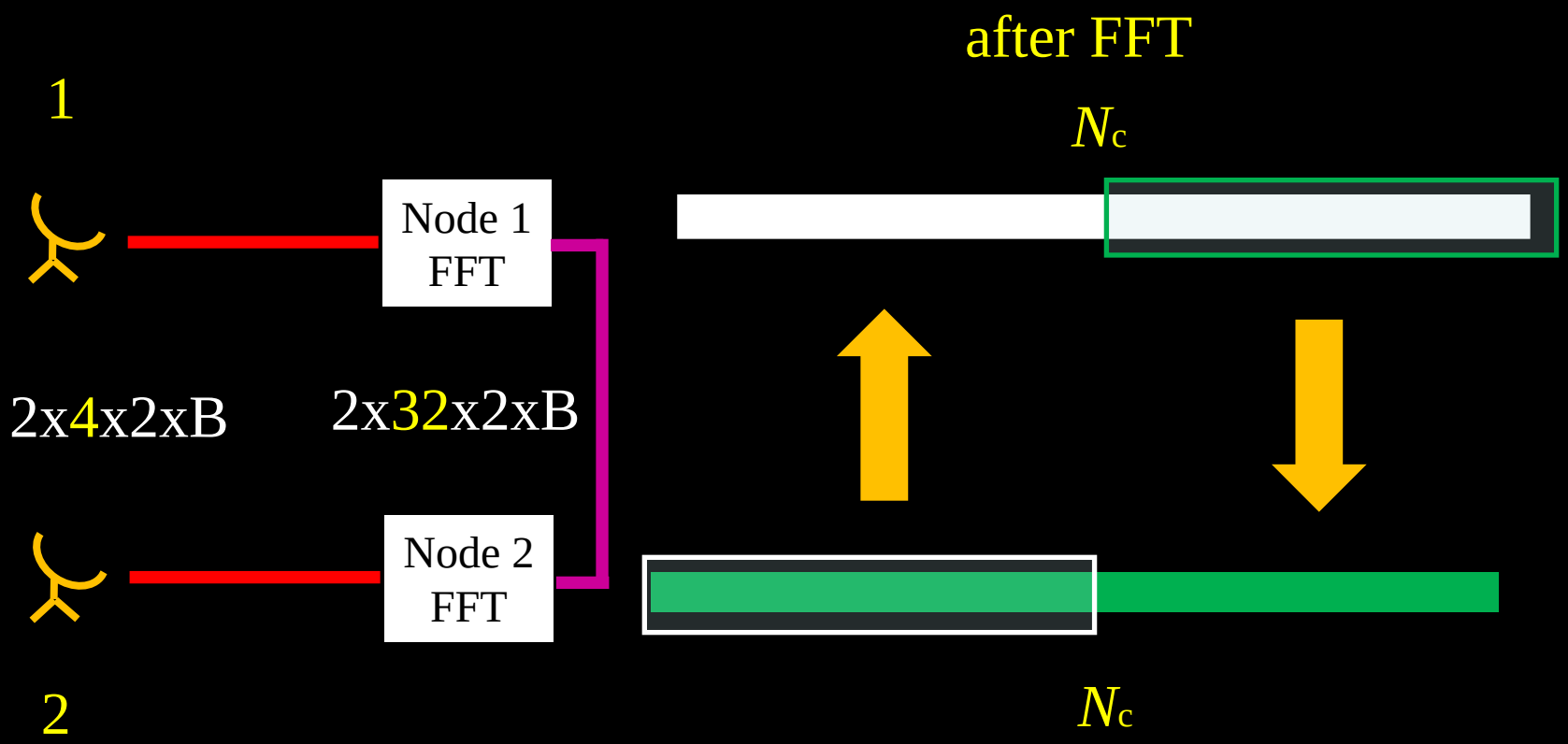
Software Correlator with current technology



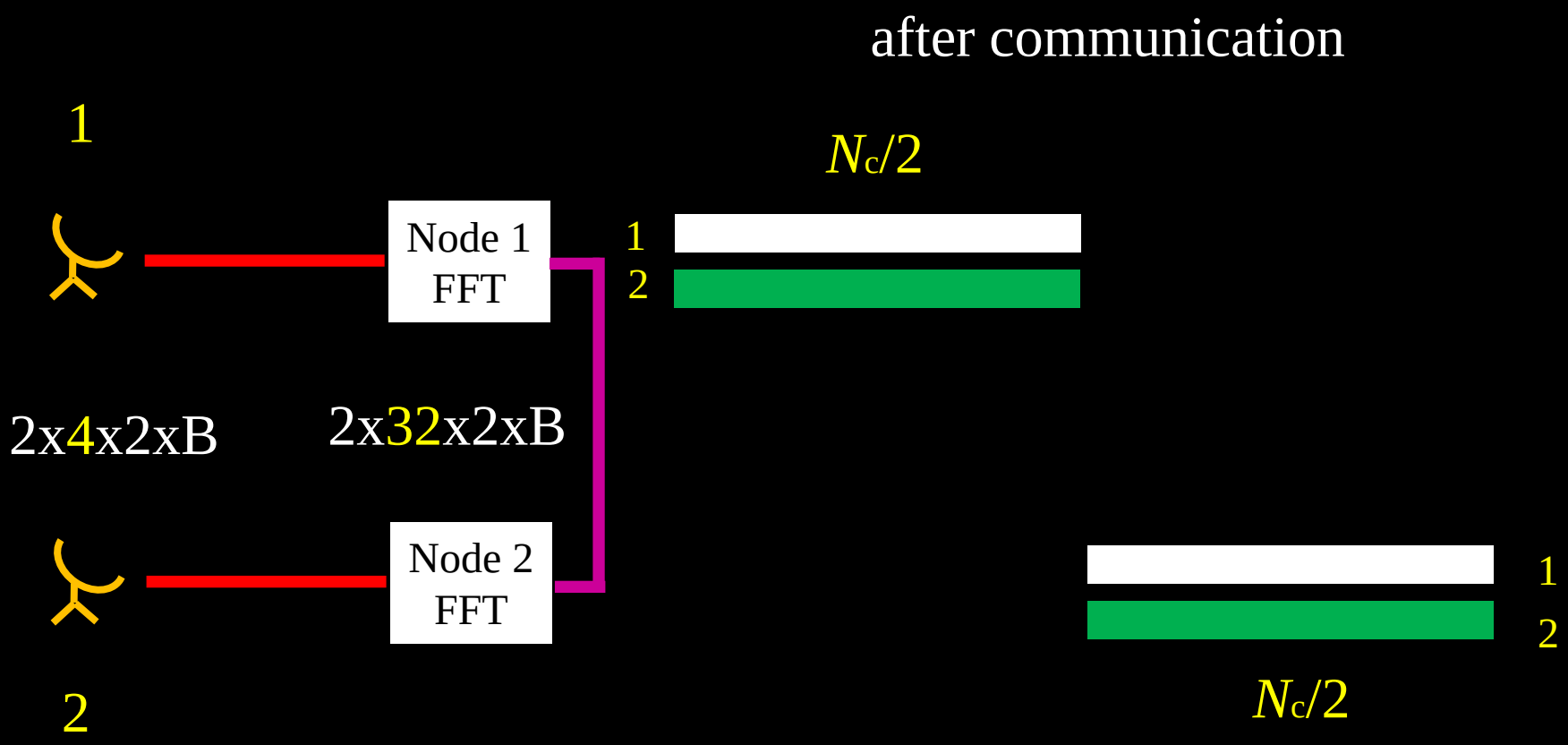
Data Flows


Node Interconnect BW:
40Gb/s (Infiniband)
CPU, FFT
4→32bits
bit/sample
GPU, CMAC
threads
memory
CPU, FFT
4→32bits
bit/sample
GPU, CMAC
threads
memory
GPU Memory
BW:
102GB/s (C1060)
PCI-e BW: 8GB/s (gen2 16x)

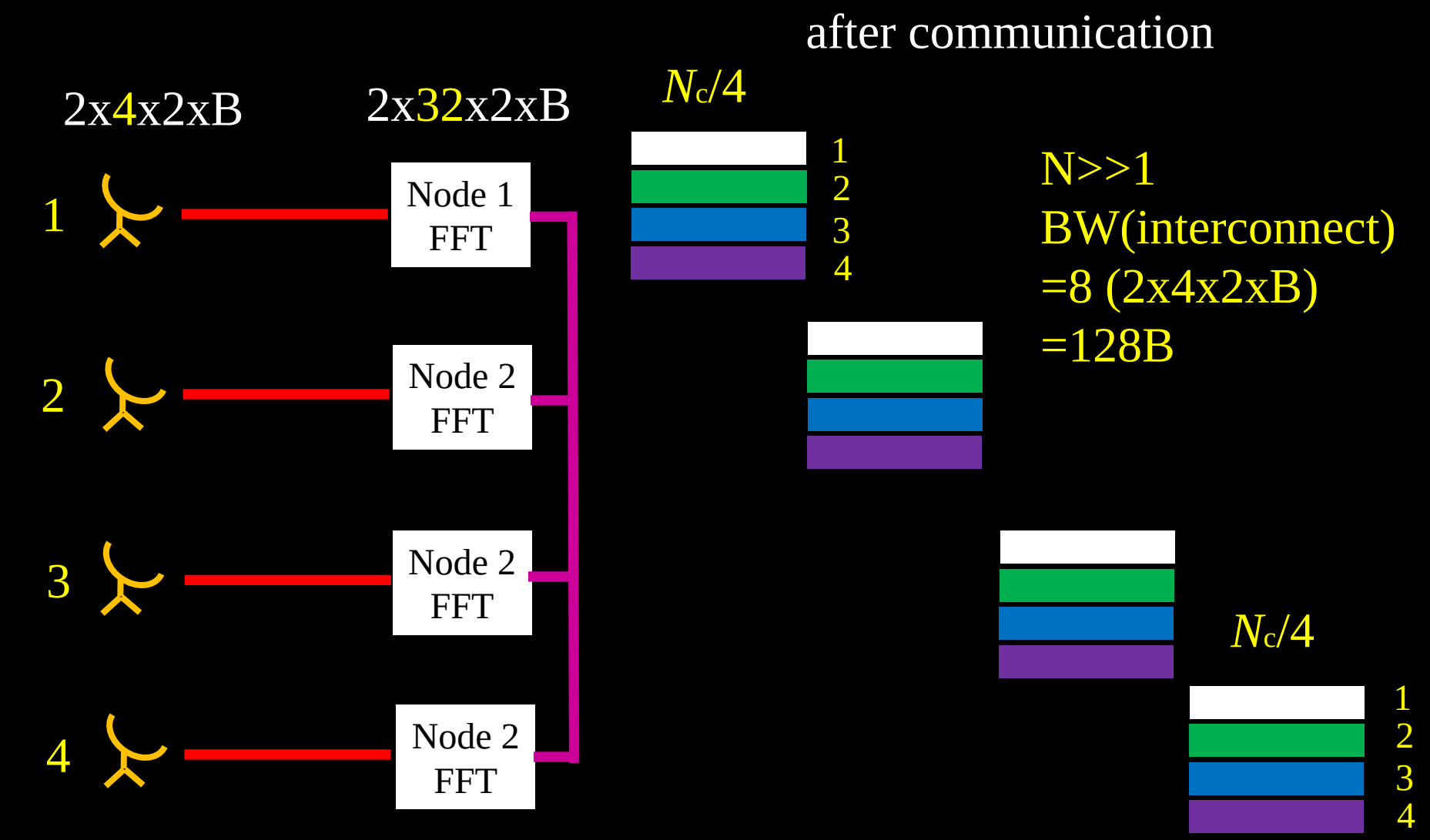
Communication between Computer Nodes I



Communication between Computer Nodes II



All-to-All Communication between Computer Nodes III



AI (Arithmetic Intensity)

- Definition: number of operations (flops) per byte
- AI = 8flops/16bytes (R_i,R_j) = 0.5

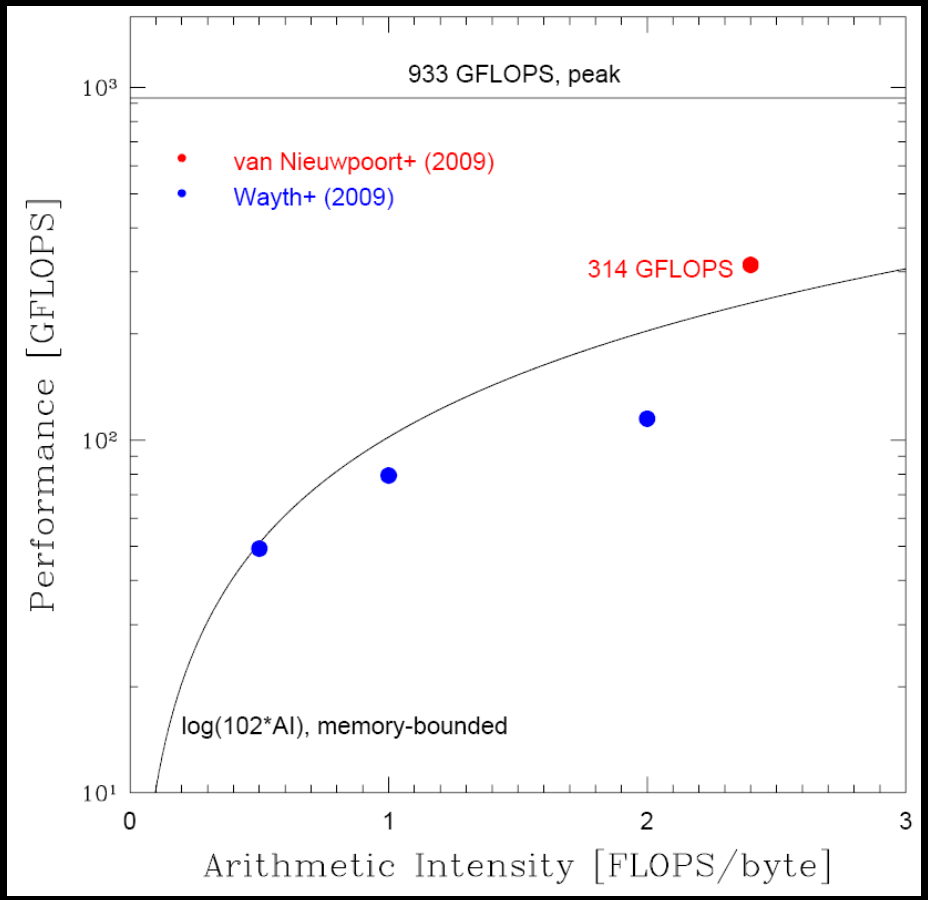
$$+R_i R_j^* = +(a_i + ib_i)(a_j - ib_j) = +(a_i a_j + b_i b_j) + i(b_i a_j - a_i b_j)$$

  AI = 32 flops/32bytes (R_i, L_i, R_j, L_j) = 1.0 for 1x1 tile

  AI = 2.4 for 3x2 tiles

- Since AIs are small numbers, correlation calculations are bounded by the memory bandwidth.
- Performance: AI x memory BW (=102GB/s)

Wayth+ (2009), van Nieuwpoort+ (2009)

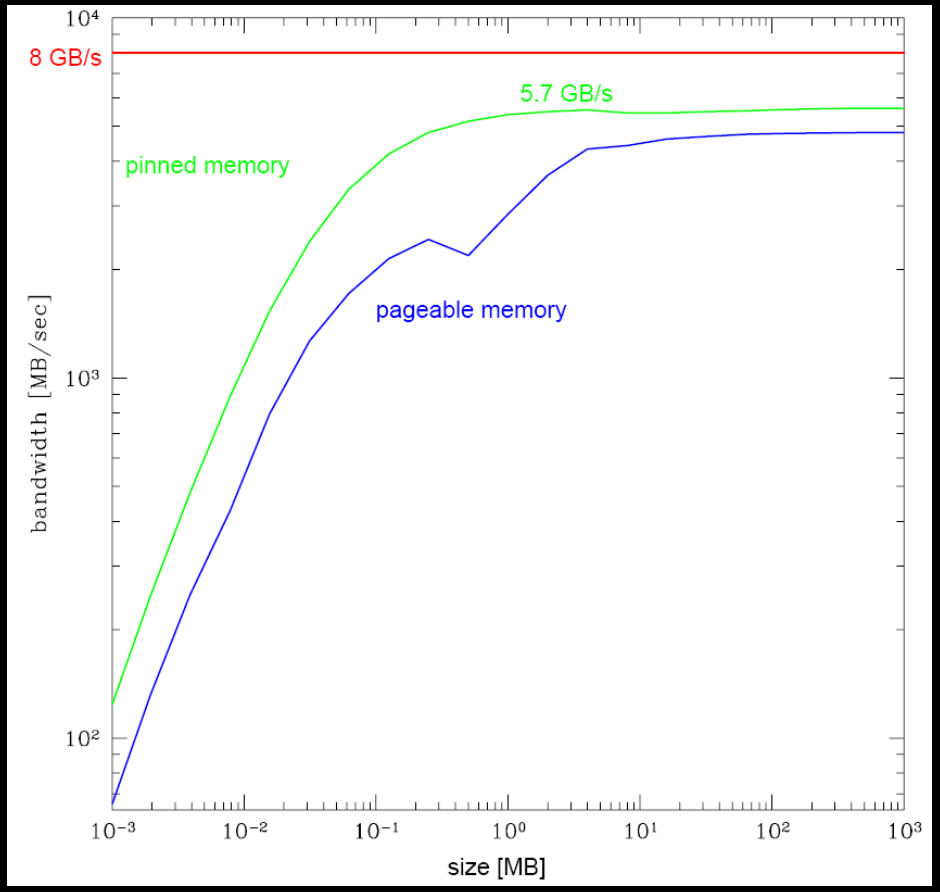


Performance of Tesla C1060 as a function of AI

- Performance is, indeed, memory-bounded.
- Maximum performance is about 1/3 of the peak performance.

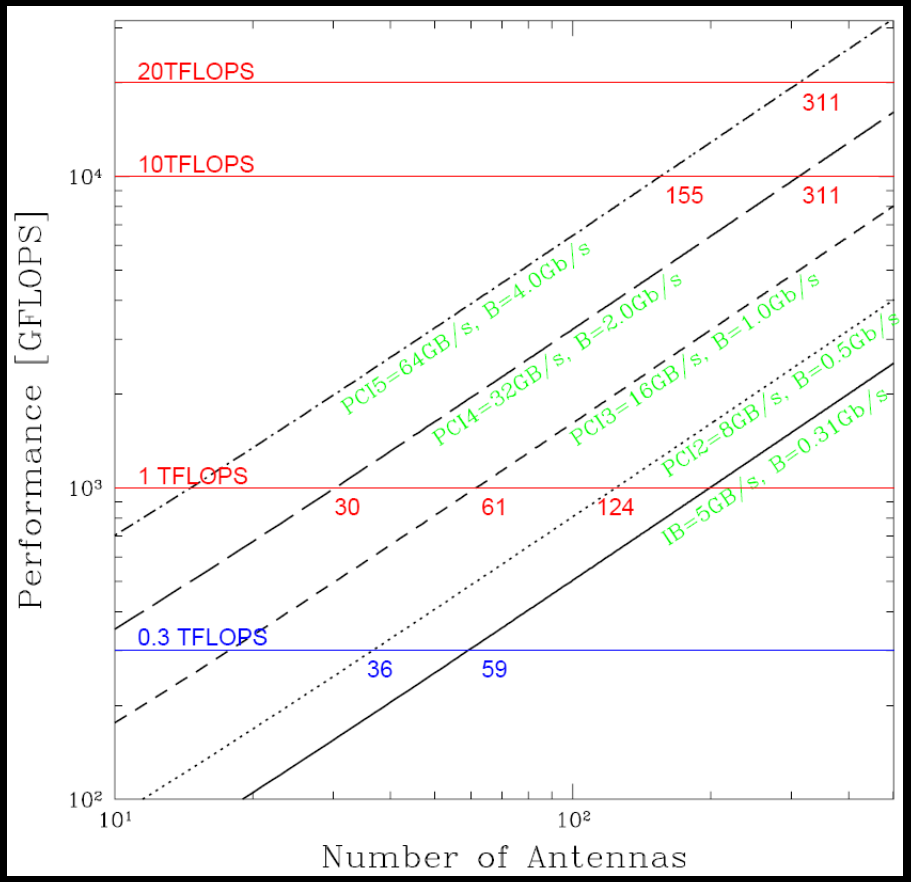
AI for host-device and host-host

- AI = N+1 FLOP/byte
  N x 16 bytes (R,L) = 16 N byte
  4x8xN(N+1)/2 FLOP
- PCI bus bandwidth
  PCI-e 2.0: 8.0GB/s (15 Jan. 2007)
  PCI-e 3.0: 16.0GB/s (2Q 2010, 2011)
  PCI-e 4.0: 32.0GB/s (201?)
  PCI-e 5.0: 64.0GB/s (201?)
- Performance [GFLOPS] =
  PCI BW [GB/s] x AI [FLOP/B]



Measured bandwidth of host-to-device (Tesla C1060)

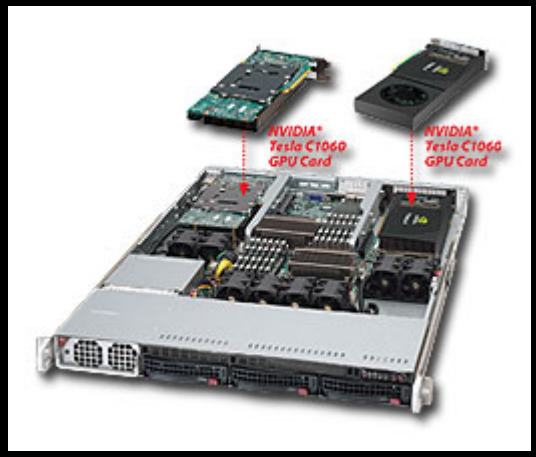
~70% of PCI -e2 bandwidth



Expected performance bounded by the BWs of PCI bus and interconnect

Power usage and Costing

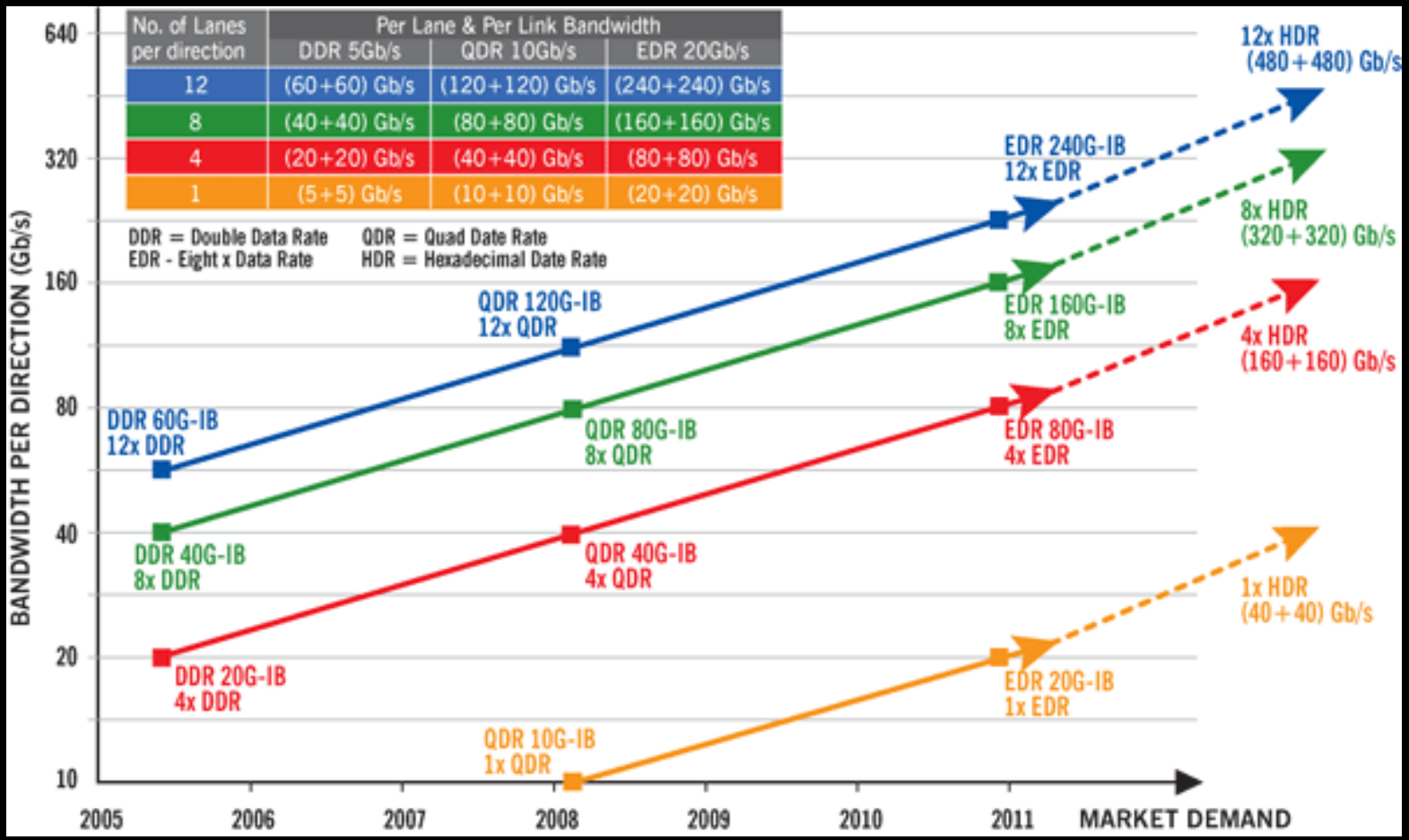
- Computer nodes
  - 1.4 KW, 4 K Euro for each server including 2x0.236 KW (2 GPUs)
  - 0.4 MW, 1.2 M Euro for 300 servers
- Network Switches
  - 3.8 KW for IB (40Gb) 328 ports



Technology Development in 2010

- 2Q 2010, (2011): PCI-e 3rd Generation
- 2Q (April) 2010: Nvida Fermi (512 cores, L1,L2 cache)
- (March 29) 2010: AMD 12 core Opteron 6100 processor
- (March 30) 2010: Intel 8 core Nehaem-EX Xeon processor

Infiniband Roadmap (IBTA)



Software Correlator for Phase I SKA (‘2018)