Mitglied der Helmholtz-Gemeinschaft

JÜLICH
FORSCHUNGSZENTRUM

Using Blue Gene Active Storage for Neuro-science Applications

D. Pleiter | CODE JAM 2014 | 27 January 2014

Outline

Blue Gene Active Storage (BGAS) architecture

- Overview on the architecture
- Motivation for architectural approach

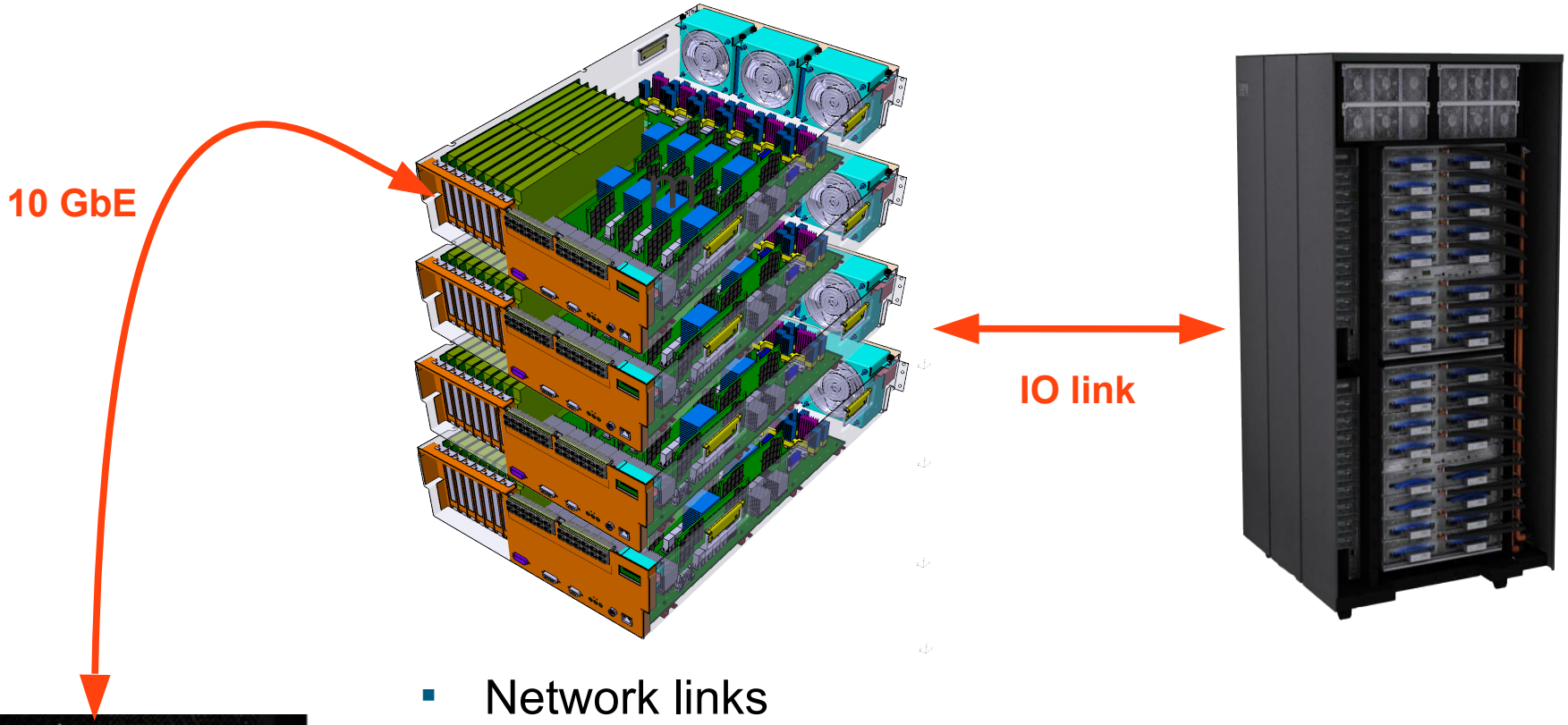
Use cases

- Categorization and selected examples

Programming models

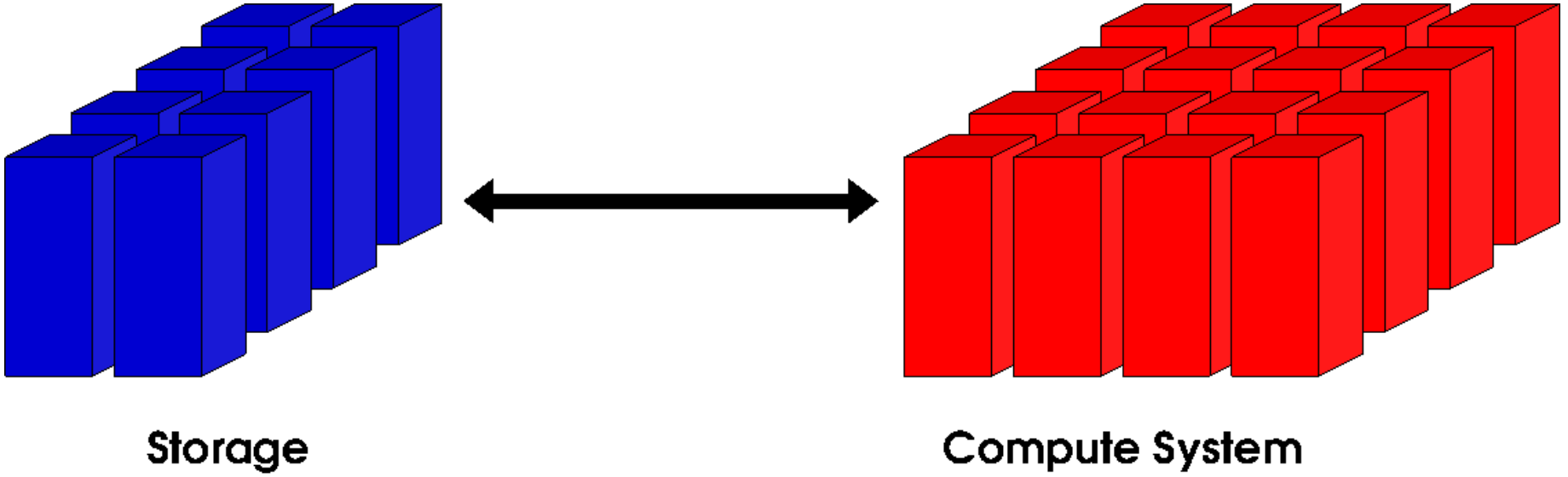
- Models for how to program an active storage architecture

Blue Gene/Q I/O architecture

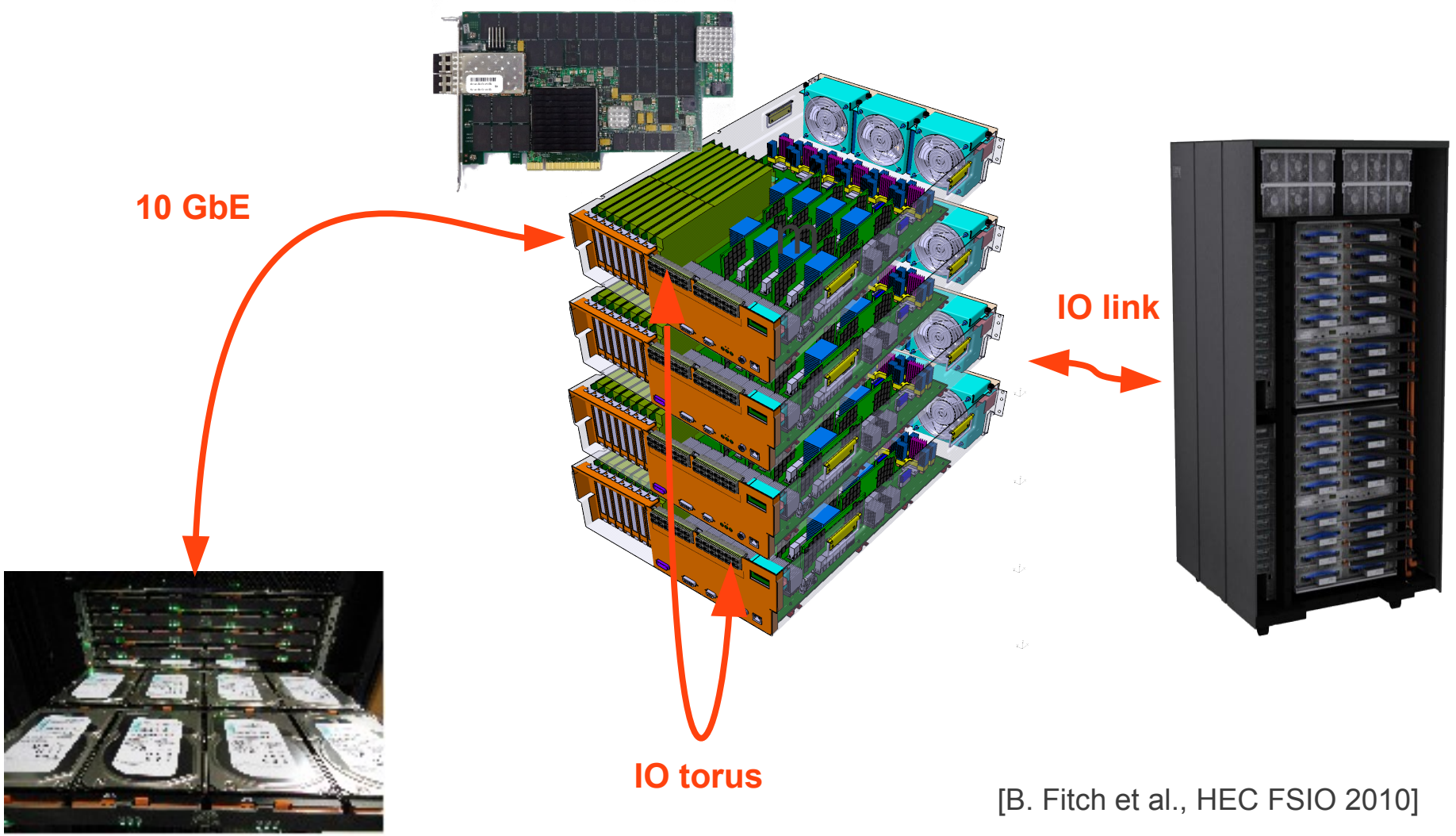


- Network links
  - 2 compute nodes connected to 1 I/O node
  - 2 10 GbE interfaces attached to 1 I/O node
- Forwarding of I/O functions
  - Slimline OS on compute nodes: CNK
  - Full Linux running on I/O nodes

Blue Gene/Q I/O architecture (cont.)

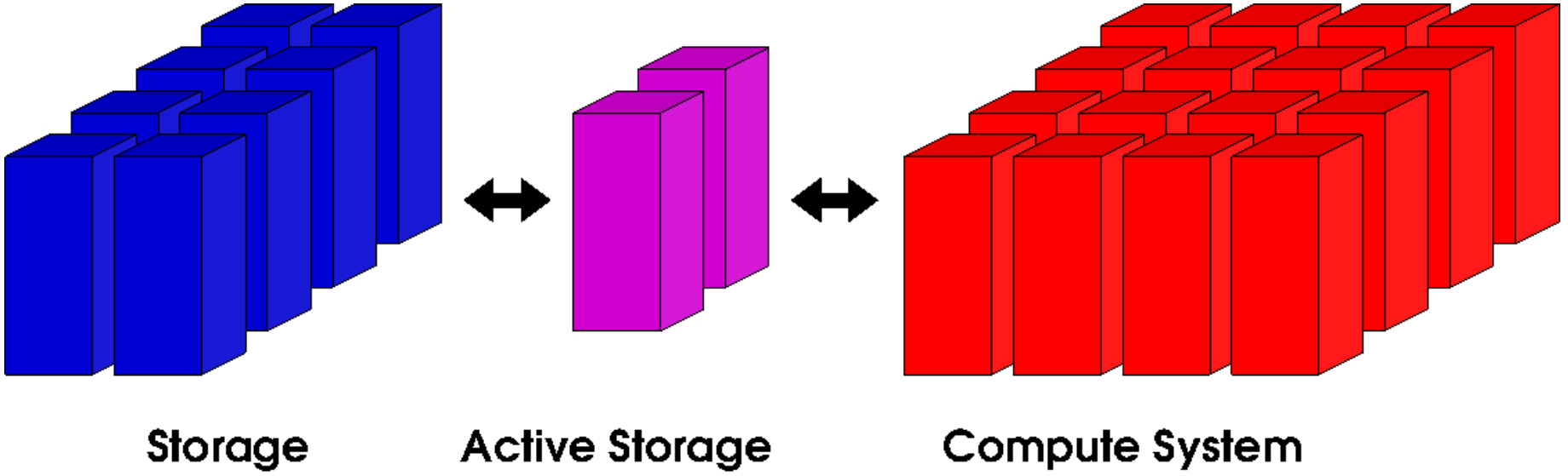


Blue Gene Active Storage

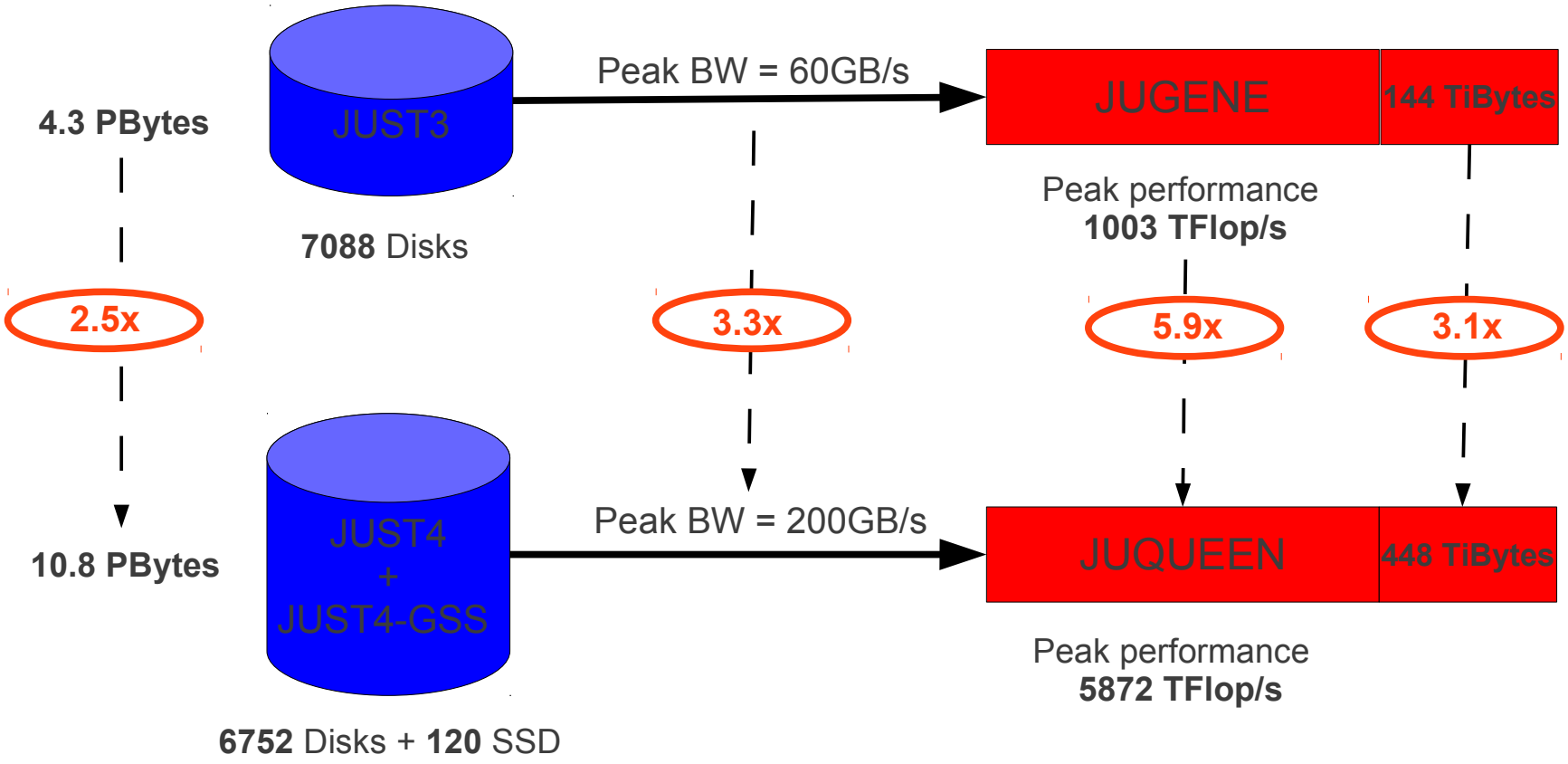


[B. Fitch et al., HEC FSIO 2010]

Blue Gene Active Storage (cont.)



JUGENE → JUQUEEN performance



Digression: Solid state storage

Technologies

- Today = NAND Flash memory
  - For HPC: SLC NAND flash
- Future= MRAM, PCM, … ?

Advantages

- High bandwidth
- Low latency → high IOPS
- Power efficiency
- High density → packaging/integrate-ability

Disadvantages

- Limited endurance (~100k write cycles for SLC)
- High costs

Digression: Active storage concepts

Data processing is cheap

- High computational densities thanks to Moore's “law”

Data transport is expensive

- Expensive in terms of energy costs

Processing capabilities close to data

- Active storage = Tight integration of compute devices and storage devices
- Concept originally introduced as “active disk”
  Anurag Acharya et al., “Active Disks: Programming Model, Algorithms and Evaluation” (1998)

BGAS: Data transport

I/O links

- Connect 2 compute nodes to 1 I/O node
- 8-32 (or more) links per rack

Torus network links

- Compute nodes and I/O nodes are interconnected by 5- and 3-dimensional network

Ethernet links

- 2 10-GbE ports per I/O node

PCIe links

- 8x PCIe GEN2 links connect I/O node and HS4 card
- 2+2 GByte/s bandwidth for flash memory and Ethernet ports

BGAS in Jülich

32 BGAS nodes

- Each node comprises 1 HS4 card
- 64 TByte nominal storage capacity

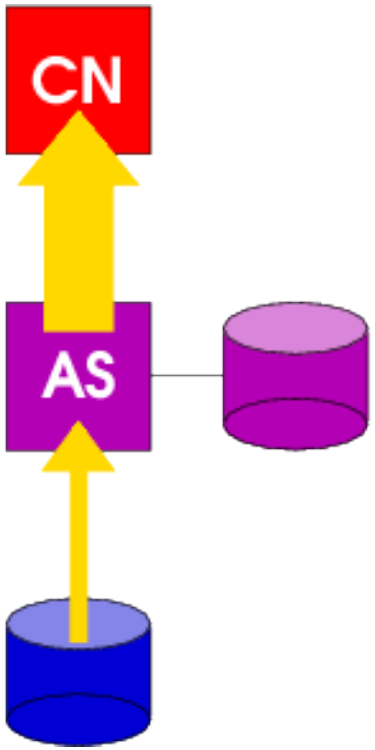
2048 compute nodes

- Part of JUQUEEN production system

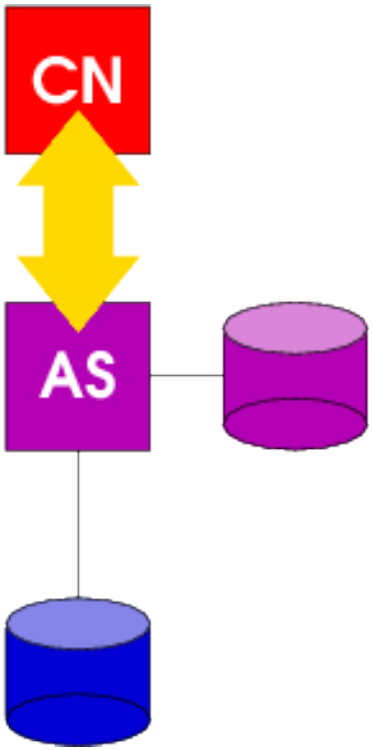
Comparison

	JUST4	BGAS
Peak performance compute [TFlop/s]	5,872	419
I/O link bandwidth [GByte/s]	496	128
I/O bandwidth [GByte/s]	200	64+64
Capacity [PiByte]	11	0.06
Peak performance active storage [TFlop/s]	--	6.6
Bi-section bandwidth [GByte/s]	--	64

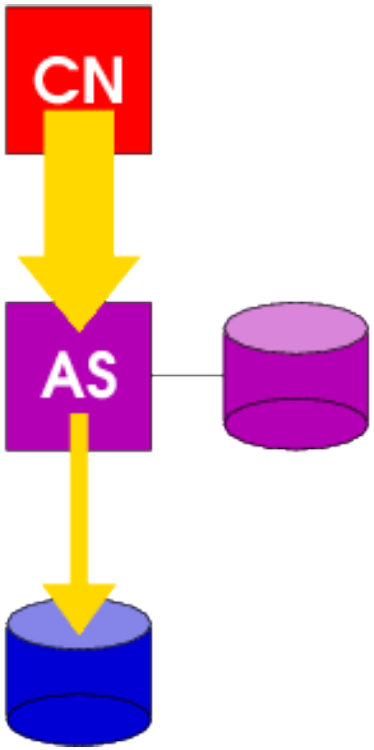
Use case categorization: Data flow



Multi-pass analysis



Out-of-core



Post-processing / Visualisation

Use cases in computational neuro-science

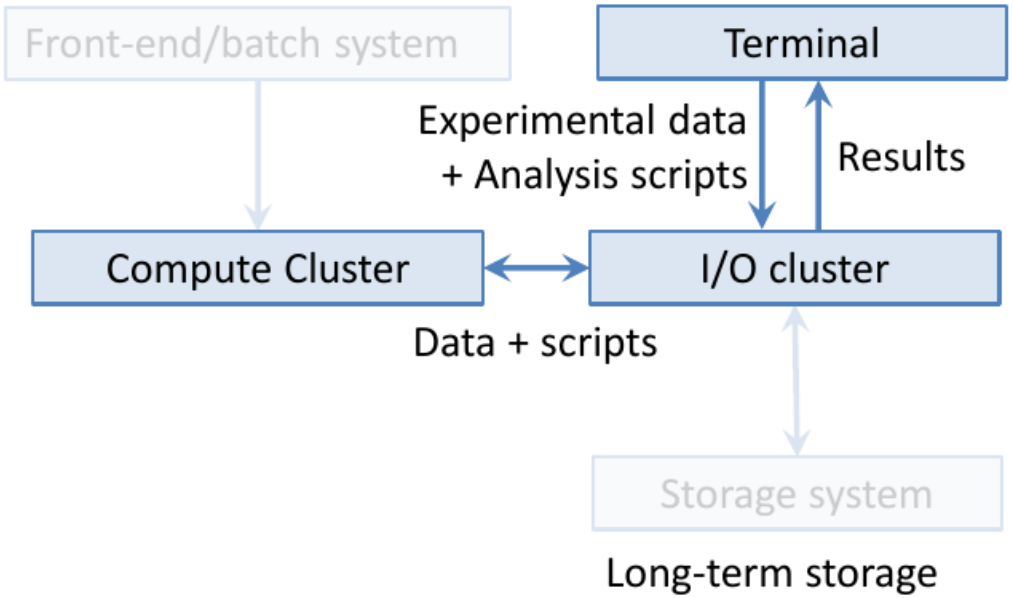
[J. Eppler]

BGAS provides close-to-local storage

- Analysis can partly be done online
- Data does not have to be moved to external storage

Envisioned use:

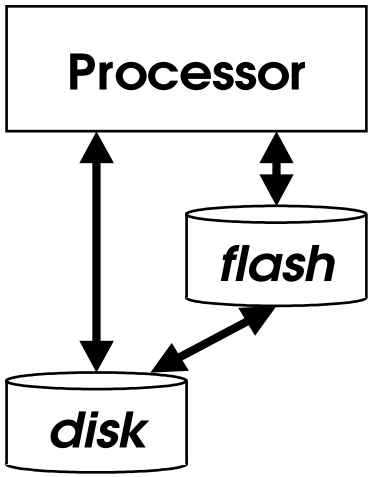
- Buffering of simulation data
- Run post-processing and data analysis
- Data analysis using compute system as accelerator



Use model: Communication via file-system

File system on flash memory

- Options:
  - Local file system
  - Parallel file system
- Flash memory used as storage device



Advantages

- File system = well-known and accepted abstraction of storage
- Easy to implement

Disadvantages

- Additional software overhead
- Data locality difficult to exploit in case of parallel file system

Library support

- SIONlib

Use model: Active messages

Concept

- Messages forwarded from compute to BGAS system get processed on BGAS system
- Message = function reference + data
  - Referenced function executed asynchronously by BGAS nodes
- Flash memory used for transient buffer space

Advantages

- Simple model, decouples simulation and data post-processing

Disadvantages

- No established interfaces

Use model: Coupled applications

Architecture viewed as coupled parallel clusters

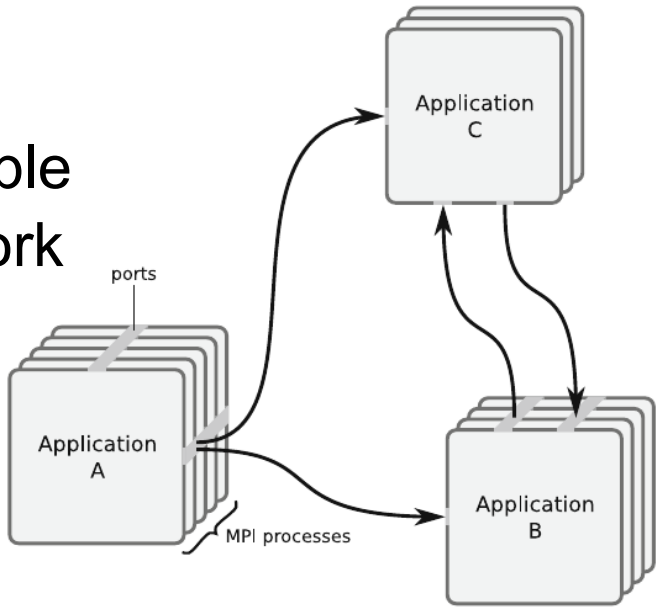
- Blue Gene/Q: Good for compute-intensive applications
- BGAS: Optimized for data-intensive applications

Concept

- Start parallel MPI applications on both clusters
- Provide send/receive operations for communication between both applications

Discussion

- No generally accepted interfaces available
- Ansatz compatible with MUSIC framework [Djurfeld, 2010]
  - API for coupling parallel computational neuro-science applications



Summary and conclusions

New I/O architectures required

- Increasing performance gap
- Non-traditional, data-intensive HPC applications
- Power constraints

New opportunities to address I/O challenge

- Non-volatile memory technologies
- Active storage architectures

Blue Gene Active Storage

- Turns Blue Gene into an architecture suitable for data-intensive applications, e.g. computational neuro-science applications

Programming models: work in progress

- Feedback and collaboration welcome