BlueField in HPC

Experience with the DINE cluster

Alastair Basden and many others
Durham University / DiRAC

DiRAC
High Performance
Computing Facility

Durham
University
Institute for Computational
Cosmology

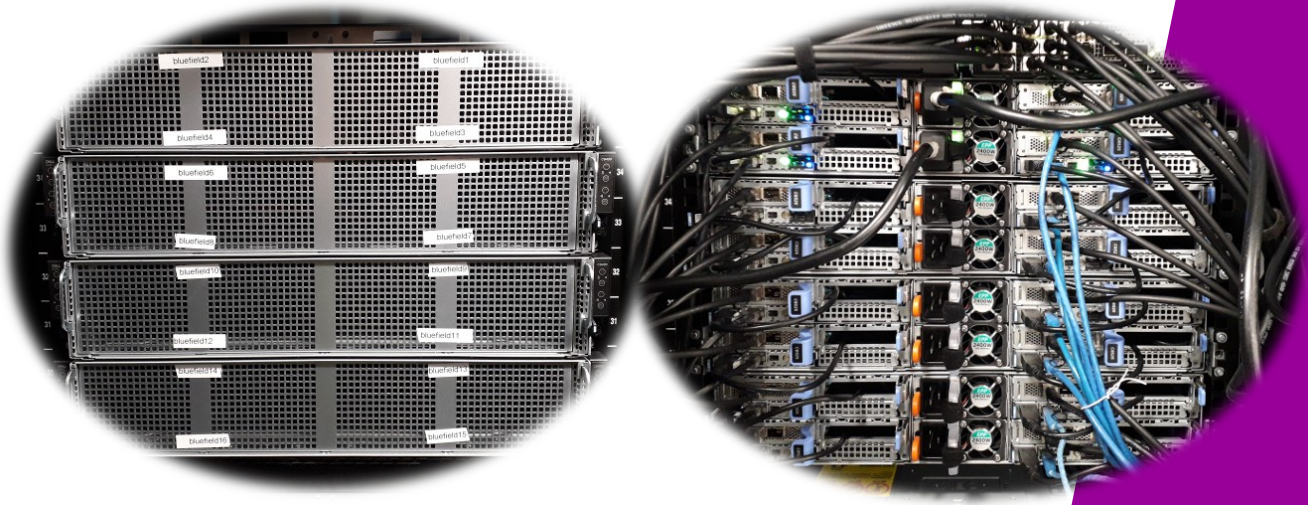
What is BlueField?

- The NVIDIA (Mellanox) Data Processing Unit
  - DPU (previously, known as an Intelligent NIC)
- Offloads data processing from a CPU
  - A programmable NIC
  - ARM cores (Linux)
  - Connect-X NIC chip
  - Hardware accelerators
  - Optional GPU chip
- Primary use cases: Not HPC
- This talk: Using BlueField in HPC)
  - Heterogeneous compute

Credit: nvidia.com

The DINE cluster

- A 16/24 node test cluster
  - Part of the COSMA DiRAC HPC facility, with Durham University investment
  - Dell C6525 half-U (double density servers)
  - 2x 16-core AMD Rome processors, 3GHz, 512GB RAM
- 16/24 BlueField cards
  - 16x BF-1, 25G Ethernet
  - 24x BF-2, 200G IB
- Available to the UK community via ExCALIBUR



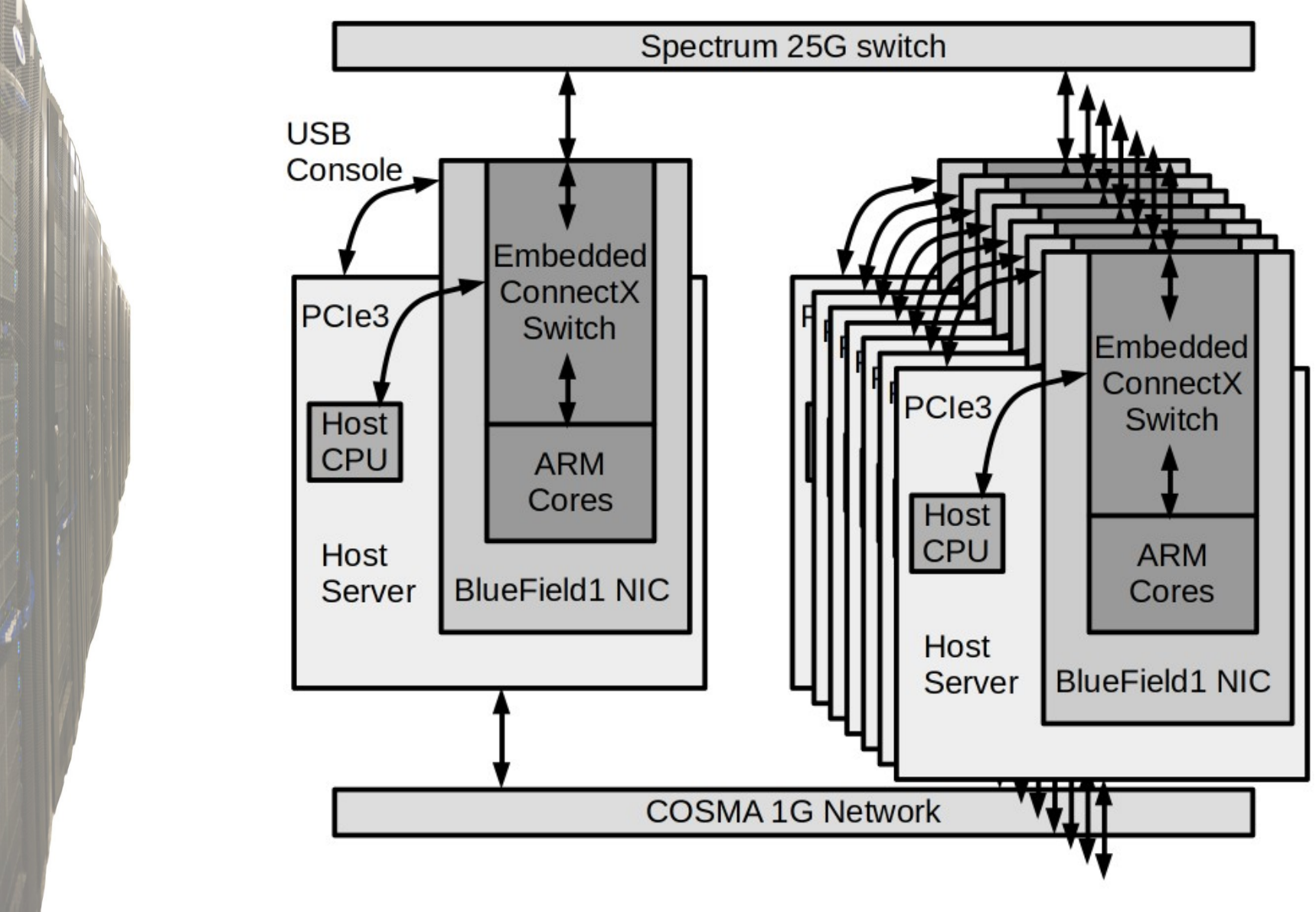
COSMA / DiRAC

- DiRAC: STFC-funded HPC Tier-1 service
- COSMA: The DiRAC Memory Intensive service
  - 4 generations of system in operation
  - COSMA8 has just come online as part of DiRAC-3
    - AMD Rome, Dell 1/2U servers, 1TB RAM/128 cores
  - ~70k cores, 13PB storage, 26PB tape

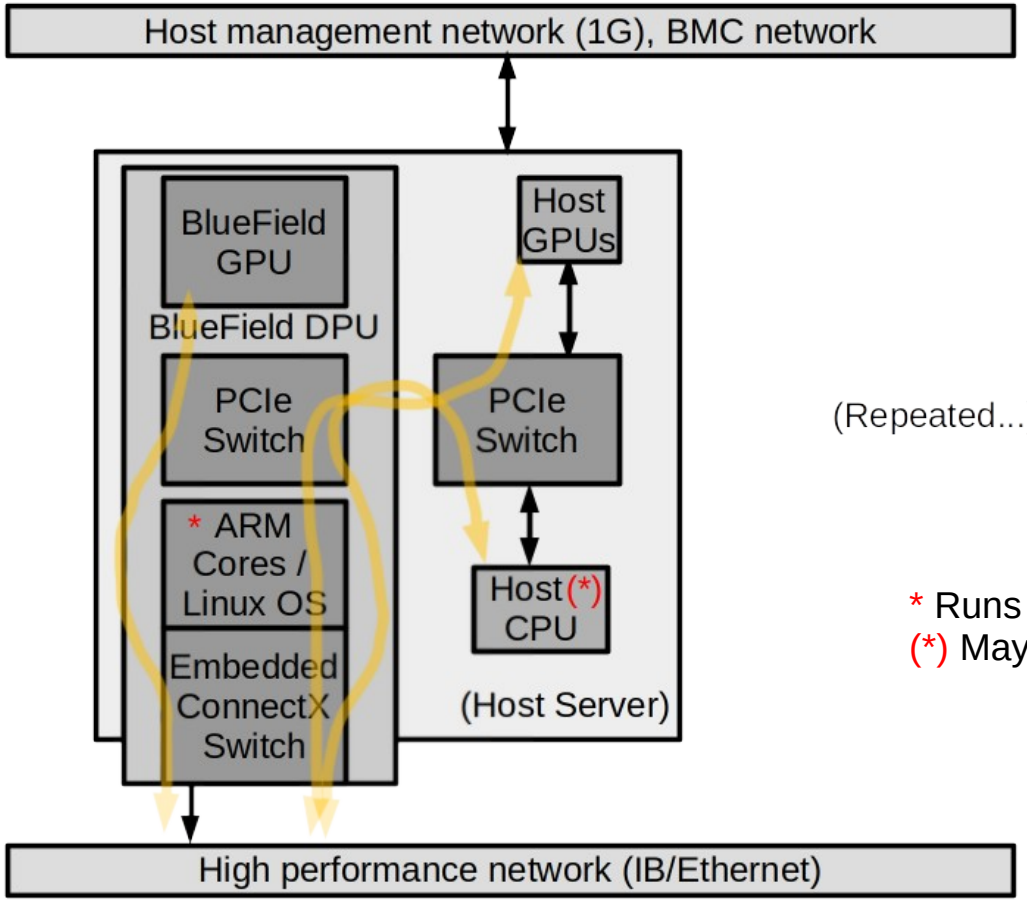
BlueField network

- Host-separated mode:
  - ARM cores run separate Linux OS
  - Have their own MAC address
    - Separate from that of the host
  - But physically share the same network interface
  - Connect-X chip routes between them
  - Look like separate devices on the network

Credit: nvidia.com


Spectrum 25G switch
USB
Console
Embedded
ConnectX
Switch
PCIe3
Host
CPU
ARM
Cores
Host
Server
BlueField1 NIC
COSMA 1G Network

Re-centring around the DPU



(Repeated...)

* Runs MPI tasks
(*) May run MPI tasks

Flexible HPC

- MPI tasks run on the BlueField ARM cores
- GPUs (on BF, or within node) can be used
- Server CPUs can be used (separate MPI tasks)
  - But this is optional: possibly to have no CPU intervention
  - Server CPUs become an accelerator
- Host RAM can be accessed from GPU or ARM
- GPUDirect can also be used
- For traditional CPU intensive tasks:
  - MPI tasks run only on the server CPU (and optionally GPUs)
- Low latency network-bound tasks:
  - MPI tasks on the DPU cores

Heterogeneous compute

- Combining ARM, X86, GPU
- Maximising the ability of BlueField for data orchestration

DPU HPC use-cases

- Data migration
- Task migration and management
- MPI progression

DPU data migration

- DPU can access host memory by RDMA
  - No host processor intervention required
  - This can be used to seamlessly move or repackage data
    - RDMA to remote host

Task migration and management

- Load balancing becomes problematic for Exascale systems
  - Some nodes become congested
  - Others are starved of tasks
- DPU can be used to migrate tasks
  - And return the results
  - Determine where new tasks should run, based on host load
- Ideal for task-based parallelism codes
  - See work on ExaHype

MPI progression

- MPI operations are not always scheduled as expected
  - The MPI progression issue
  - Compute cores can be blocked waiting for their MPI task to complete or be scheduled
    - CPU time wasted
  - Offload MPI operations to the DPU:
    - Data passed to DPU, host CPU can then continue
    - DPU performs the MPI task
    - DPU can ensure MPI progression using dedicated threads

Lessons learned

- Not an out-of-the-box solution
  - Sys-admin/resOps effort is required
- Requires effort from users
  - Have to be willing to compile ARM and x86 binaries
  - Have to give correct command line args when launching an MPI task spanning host and BF
  - Have to understand what they’re doing
  - Not all MPI libraries supported

Now for some detailed information

- Hardware installation
- Network interfaces
- Software configuration
  - Including for the BF devices
- Additional information
- Next steps

Hardware installation

- Insert the BF cards
  - 16 screws per card!
    - Removal of full height face plate
    - Removal of 1st PCIe riser
    - Removal of 2nd PCIe riser to let USB cable through

Hardware thoughts

- A bit of a faff
  - Why not put the USB port on the face plate! Or within the PCIe switch
  - Why not ship with a half-height face place!
- The USB port (serial interface) was essential
- When we later wanted to install a 2nd PCIe card, we modified the face plate to allow the USB cable through it (using some tin snips)

Network interfaces

Hostname	IP address	Host or device	Network interface	Network	Comment
b101 – b116	172.17.178.201 – 216	Host	em1	COSMA 1G	COSMA network
Unnamed	192.168.100.1 – 31 (odd)	Host	tmfifo_net0	Internal PCIe console	Low bandwidth
bluefield101 - 116	192.168.100.2 – 32 (even)	Device	tmfifo_net0	Internal PCIe console	Low bandwidth
bfh101 – 116	192.168.101.1 – 31 (odd)	Host	p1p2	25G	Dedicated Bluefield network
bfd101 – 116	192.168.101.2 – 32 (even)	Device	enp3s0f1	25G	Dedicated Bluefield network
bflocal (only from a host)	192.168.101.2 – 32 (even)	Device	enp3s0f1	25G	This hostname used to access the device within the current server

Software config

- CentOS7 on servers (standard COSMA7 image)
- BlueField drivers including rshim kernel module installed
- Internal tmfifo_net0 interface brought up
- NAT set up on hosts, iptables installed - providing file system access
- sshd config modified
- Network settings for p1p2 interface (The BlueField 25G Ethernet fabric) created
- BlueField device booted with CentOS7 image (.bfb from NVIDIA-Mellanox)
- MOFED installed, OpenIBD service restarted
- bflocal added to /etc/hosts, IP pointing to local card on the 25G network
- 25G IP addresses added to DNS (host and device)

Software config of BF devices

- Initial access:
  - via tmfifo_net0 interface (low bandwidth internal NIC)
  - screen /dev/ttyUSB0 115200, root/centos
  - Set gateway to host IP address to provide a route to rest of cluster.
- Set hostname (hostnamectl)
- Add search criteria to /etc/resolv.conf
- Add Ethernet COSMA file systems to /etc/fstab
- Install kerberos and IPA packages, and set up config
- Reboot host and device
- Enrole device in IPA, create sshd config file, restart sshd
- Switch device into host-separated mode
  - Reboot host server. At this point, the server now sees a boot problem, and requires F1. Probably some non-compliance.
- Change default gateway to the 25G network
- For RDMA, mediated device required, enable some mlxconfig flags
- Install MOFED

Other things

- Haven’t yet built our own bfb file (Arm Linux image), though we have instructions on how to do that.
- Integration of the BlueField device with cluster manager was harder because they weren’t on the same network
  - No direct access for cluster manager
  - Hence creation of a bflocal address on the hosts
    - Allowing effective pdsh access to all BlueField devices
- Some experiments with OpenVSwitch

Next steps

- Upgrade to BlueField-2
  - HDR200
  - 2-node test system almost ready
    - OOB Network interface (direct access to SLURM etc)
    - No F1 prompt on reboot!
- Closer integration with system package manager
  - Updating packages etc
- Closer integration with SLURM
  - A BlueField-only queue?
  - Mixed BlueField-Host queue?
  - Will require direct contact between BF and SLURM
    - Should be easier with BF2

Heterogeneous compute: Other stuff

- GPU acceleration of adaptive atmospheric correction system
  - First “on-sky” demonstration of GPU for telescope control worldwide
- FPGA acceleration of HPC code
  - Xilinx FPGAs tightly coupled to AMD X86 processors
    - Avoiding PCI bottleneck
  - Used in offload mode
    - Dataflow pipeline moved onto FPGA
    - Algorithms include simulation of:
      - Atmosphere, telescope optics, detectors, random noise, data processing
      - Custom Mersenne Twister generator, 2D FFTs
    - Hand-coded VHDL
    - 600x speed-up over CPU-only version

Heterogeneous compute: Not new

- GPU acceleration of adaptive atmospheric correction sys
  - First “on-sky” demonstration of GPU for telescope control wo
- FPGA acceleration of HPC code
  - Xilinx FPGA tightly coupled to processor
    - Avoiding PCI bottleneck
  - Used in offload mode
    - Dataflow pipeline moved onto FPGA
    - Algorithms include simulation of:
      - Atmospheric, telescope optics, detectors, random noise, data processing
      - Custom Mersenne Twister generator, 2D FFTs
    - Hand-coded VHDL
    - 600x speed-up over CPU-only version

2010

2005:
Cray XD1

Conclusions

- Heterogeneous compute: tread carefully
  - Make codes generic, use portable standards
    - Task-based parallelism
  - Share any work, contribute to standards, libraries etc
- DINE cluster provides a research platform for DPUs
  - Hybrid X86/Arm MPI tasks
  - Data orchestration
  - Task migration
  - MPI progression
  - Ask if you wish to use it… (via ExCALIBUR)