The Arm architecture in HPC:
from mobile phones to the Top500
The Arm architecture in HPC: from mobile phones to the Top500

Filippo Mantovani  
filippo.mantovani@bsc.es

Marta Garcia-Gasulla  
marta.garcia@bsc.es

Impact of Arm hardware from an HPC application perspective  
Austin - 2019, Sep 15
Barcelona Supercomputing Center

Supercomputing services to Spanish and EU researchers

R&D in Computer, Life, Earth and Engineering Sciences

PhD programme, technology transfer, public engagement

Spanish Government 60%

Catalan Government 30%

Univ. Politècnica de Catalunya (UPC) 10%

The Arm architecture in HPC: from mobile phones to the Top500

Austin - 2019, Sep 15
The MareNostrum 4 supercomputer

Total peak performance:

13,7 PFlops/s
The archeology of Mont-Blanc (2012-2014)

In the early days Arm-based prototypes were made out of Android development kits

- Non-HPC platforms
  - 32 bits CPUs
  - 1 Gigabit Ethernet (bridged)
  - With several slow I/O interfaces

- Targeting embedded market
  - Not prepared for 24/7 operation
  - With cooling issues
  - With form factor issues

The first Mont-Blanc prototype (2015 - 2019)

Exynos 5 compute card
- 2 x Cortex-A15 @ 1.7GHz
- 1 x Mali T604 GPU
- 2 GB of LPDDR3 RAM
- 1 Gb Ethernet
- 15 Watts

Mont-Blanc rack
- 8 BullX chassis
- 72 Compute blades
- 1080 Compute cards
- 2160 CPUs
- 1080 GPUs
- 4.3 TB of DRAM
- 17.2 TB of Flash

The "Dibona" Mont-Blanc prototype (2018)

Based on Bull Sequana HPC infrastructure (both x86 and Intel)

Dibona node
2 x ThunderX2 processors (CN9980-2000LG4077-Y21-G 32 2.0)
64 x Marvell’s cores @ 2.0 GHz
256 GB memory
256 GB local storage (+ 8TB via NFS)

Dibona rack
48 x Dibona nodes
Fat tree interconnect topology
Infiniband EDR 100Gb/s
Theoretical peak: 49 TFlops

Micro-benchmarks

Memory, Computational Throughput, Network
Memory bandwidth and latency

![Graph showing memory bandwidth and latency](image)

- **Dibona**: DDR4-2666 x8 x2: 341.4 GB/s
- **MareNostrum4**: DDR4-3200 x6 x2: 307.2 GB/s

**Bandwidth [GBytes/sec]**

- **218.40 GB/s - 64%**
- **195.22 GB/s - 57%**
- **171.89 GB/s - 56%**

**Latency [ns]**

- **L1: 32 KiB**
- **L2: 256 KiB**
- **L3: 32 MiB**
CPU floating point throughput

<table>
<thead>
<tr>
<th></th>
<th>Dibona</th>
<th>MareNostrum4</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPU SP</td>
<td>7.99</td>
<td>8.36</td>
</tr>
<tr>
<td>FPU DP</td>
<td>7.99</td>
<td>8.36</td>
</tr>
<tr>
<td>SIMD SP</td>
<td>31.95</td>
<td>131.67</td>
</tr>
<tr>
<td>SIMD DP</td>
<td>15.97</td>
<td>65.79</td>
</tr>
</tbody>
</table>
Putting all together into a “roofline model”

Performance [GFlops] vs. Computational intensity [Flops/byte]

- FMA 512
- FMA 1536
Network: Infiniband bandwidth and latency

IB EDR / OPA: 12.5 GB/s

Cray Aries: 10.2 GB/s
Network: sendrecv congestion and weak links

IB EDR: 12.5 GB/s

Aggregated Bandwidth [GBytes/sec]

MPI Processes

8 B
64 B
1 KiB
8 KiB
64 KiB
512 KiB
4 MiB
Classical HPC benchmarks

LINPACK and HPCG
High-Performance LINPACK

Source: https://www.top500.org/statistics/efficiency-power-cores/
High-Performance Conjugate Gradient (HPCG)

• Relevant for the HPC community
• Not fully explored on Arm architecture
• Part of the Student Cluster Competition 2017 and 2018
HPCG shared memory implementation - 2

MPI-only strong scaling (4 iterations) on KNL

64 MPI, 10% in MPI calls

128 MPI, 20% in MPI calls

192 MPI, 40% in MPI calls
HPCG shared memory implementation - 3

- B0 silicon, OpenMPI 3.0, GCC 7.1.0, Arm HPC Compiler 18.1
- Code released [https://gitlab.com/arm-hpc/benchmarks/hpcg](https://gitlab.com/arm-hpc/benchmarks/hpcg)

Applications

Complex codes used in data-centers for production of scientific results
TensorFlow: HPC clusters evaluation

- TensorFlow v1.11
- Linear algebra using the built in library (Eigen)
- CPU only comparison
- Node-to-node comparison

![Bar chart comparing sustained performance of different systems.](chart.png)

- AlexNet
- ResNet-50

- MareNostrum 4
- Power 9
- Dibona
TensorFlow: Performance improvements targeting Arm

- Other vendors leverages optimized linear algebra libraries (e.g., MKL)
- We plugged Arm Performance Libraries into TensorFlow (proof of concept)
- Handed over to Arm: https://gitlab.com/arm-hpc/packages/wikis/packages/tensorflow

A real CFPD problem: Alya

Alya for **physicists**

- It can simulate several physics models:
- Simulations can combine multiple models (multi-physics)

Alya for **computer scientists**

- Fortran code (500 kLines)
- MPI only (production)
- Shared memory with OpenMP (experimental)
- Unstructured and hybrid meshes
- Scalability up to 100 kCores
Alya: production scientific code in action

What happens if you run a production simulation on a real HPC cluster?

Despite the effort in preparing the grid we can notice a macroscopic load imbalance.
Protect the performance

- Imbalance
- Noise
- System hardware
- System software
Protect the performance

- Imbalance
- Noise
- System hardware
- System software
Alya: Multidependences

- **Goal:** To avoid the race condition between two threads updating elements of the grid sharing variables.

```
Elements  Parallelization

Atomics
omp parallel do
omp atomic

Coloring
omp parallel do
omp parallel do
omp parallel do
omp parallel do
omp parallel do

Multidependences
omp task
mutexinoutset(iterator)
```

---

OpenMP

---

27
Alya: Multidependency evaluation

- GCC + OmpSs
- Assembly phase only (no Subgrid scale)
- Impact of atomic implementation across different architectures
- Impact of coloring across different architectures (linked to micro-architecture of caches)
Alya: Dynamic Load Balance
Alya: MPI only vs DLB evaluation

- Execution time of one “time-step”
- 2 nodes comparison
- GCC + OmpSs

Same cluster age, similar overall performance!
Alya: energy to solution [kJ] with production applications

<table>
<thead>
<tr>
<th>Nodes</th>
<th>Armv8 Cavium ThunderX2</th>
<th></th>
<th></th>
<th>x86 Skylake 8176</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GCC</td>
<td>GCC+DLB</td>
<td>Arm CLANG</td>
<td>GCC</td>
<td>GCC+DLB</td>
<td>ICC</td>
</tr>
<tr>
<td>1</td>
<td>90.17</td>
<td>68.74</td>
<td>63.44</td>
<td>101.12</td>
<td>82.15</td>
<td>69.24</td>
</tr>
<tr>
<td>2</td>
<td>96.09</td>
<td>76.56</td>
<td>69.09</td>
<td>112.52</td>
<td>84.69</td>
<td>75.54</td>
</tr>
<tr>
<td>4</td>
<td>155.40</td>
<td>78.80</td>
<td>86.34</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>204.87</td>
<td>96.05</td>
<td>111.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>230.86</td>
<td>123.34</td>
<td>157.27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>271.64</td>
<td>184.51</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- OmpSs + GCC + DLB best option for scaling
- Arm CLANG on Dibona more energy efficient than ICC on x86
- We cannot run DLB with Arm CLANG / ICC for vendor specific limitations
Worry about the programmer not about the architecture!

The Arm architecture in HPC: from mobile phones to the Top500

Austin - 2019, Sep 15
OpenFOAM: Dibona vs MareNostrum4
Dibona highlights: overall strong scalability

The Arm architecture in HPC: from mobile phones to the Top500

Austin - 2019, Sep 15
Education

The Student Cluster Competition “adventure”
The Student Cluster Competition: 5 years with Arm!

**Competition rules**
- Six undergraduate students
- One cluster managed by the team
- 3 kW power limit

**Competition awards**
- Best Linpack performance
- 1st, 2nd and 3rd overall
- Fan favorite

**The challenge**
- HPC Benchmarks
  - HPL, HPCG
- Real scientific applications
  - Quantum Espresso, TensorFlow
- Judge interviews
- Secret challenges
  - Secret application, Power outage

The Student Cluster Competition: 5 years with Arm!

Competition rules
• Six undergraduate students
• One cluster managed by the team
• 3 kW power limit

Competition awards
• Best Linpack performance
• 1st, 2nd and 3rd overall
• Fan favorite

The challenge
• HPC Benchmarks
  – HPL, HPCG
• Real scientific applications
  – Quantum Espresso, TensorFlow
• Judge interviews
• Secret challenges
  – Secret application, Power outage

---

We are looking for a cluster for ISC 2020!

SCC with Arm: a successful educational package

Effort
• ~160h of work
  Amounts to six ECTS
  European Credit Transfer System
• ~530h of cluster usage
• 23 students (and counting!)

Cluster and other hardware resources
• Supported by sponsors
• Absorbed by university

Daily cost when traveling
• ~100 EUR/day per student

~80% of students actively engaged in HPC after being part of the Barcelona SCC team

SCC with Arm: a successful educational package

Effort

• ~160h of work
  Amounts to six ECTS
  European Credit Transfer System

• ~530h of cluster usage

• 23 students (and counting!)

Cluster and other hardware resources

• Supported by sponsors

• Absorbed by university

Daily cost when traveling

• ~100 EUR/day per student

We are STILL looking for a cluster for ISC 2020!

~80% of students actively engaged in HPC after being part of the Barcelona SCC team

Conclusions
And what’s next?

Research on SoC architecture for next generation HPC systems

European roadmap to Exascale for data-centers and automotive

Performance Optimization and Productivity
• Promoting best practices in performance analysis and parallel programming
• Precise understanding of application and system behavior
• Suggestion/support on how to refactor code in the most productive way
• Transversal across application areas, platforms, scales
• For academic AND industrial codes and users
https://pop-coe.eu/


Student Cluster Competition and Education


In case you liked the talk and you want to follow up:

- Speaker: filippo.mantovani@bsc.es

- More info about Mont-Blanc:
  - https://www.montblanc-project.eu/
  - https://twitter.com/Mont Blanc_Eu
  - https://www.linkedin.com/company/mont-blanc-project/

- Visit BSC, the most beautiful datacenter in the world, if you come to Barcelona

Acknowledgment: The Mont-Blanc application team at BSC