# Arm HPC Solution

A partner enabled ecosystem

Srinath Vadlamani

8th NCAR MultiCore Workshop (MC8), Boulder CO, Sept 18019, 2018

# HPC Strategy

**Mission:**
Enable the world’s first Arm supercomputer(s)

**Strategy:**
Enablement + Co-Design + Partnership

## Building Blocks

### Enablement

- Address gaps in computational capability and data movement within Architecture
- Seed the software ecosystem with open source support for Armv8 and SVE libraries, tools, and optimized workloads
- Provide world class tools for compilation, analysis, and debug at large scale.

### Co-Design

- Work with key end-customers in DoE, DoD, RIKEN, and EU to design  balanced architecture, uArchitecture and SoCs based on real-world workloads, not benchmarks.
- Develop simulation and modelling tools to support co-design development with end-customers, partners, and academia.

### Partnership

- Work with Architecture partners to bring optimized solutions to market quickly.
- Work with ATG &  uArchitecture design teams to steer future designs to be more relevant for HPC, HPDA, and ML
- Work with key ISVs to enable mid-market

# Software Tools

arm

# But to engage with customers, we need a reason…

## Arm Allinea Studio

Develop and run on today’s hardware

| Arm Compiler for HPC | Arm Performance Libraries | Arm Forge Professional | Arm Performance Reports |
|---|---|---|---|
| Linux user space compiler for HPC applications | BLAS, LAPACK and FFT | Multi-node interoperable profiler and debugger | Interoperable application performance insight |

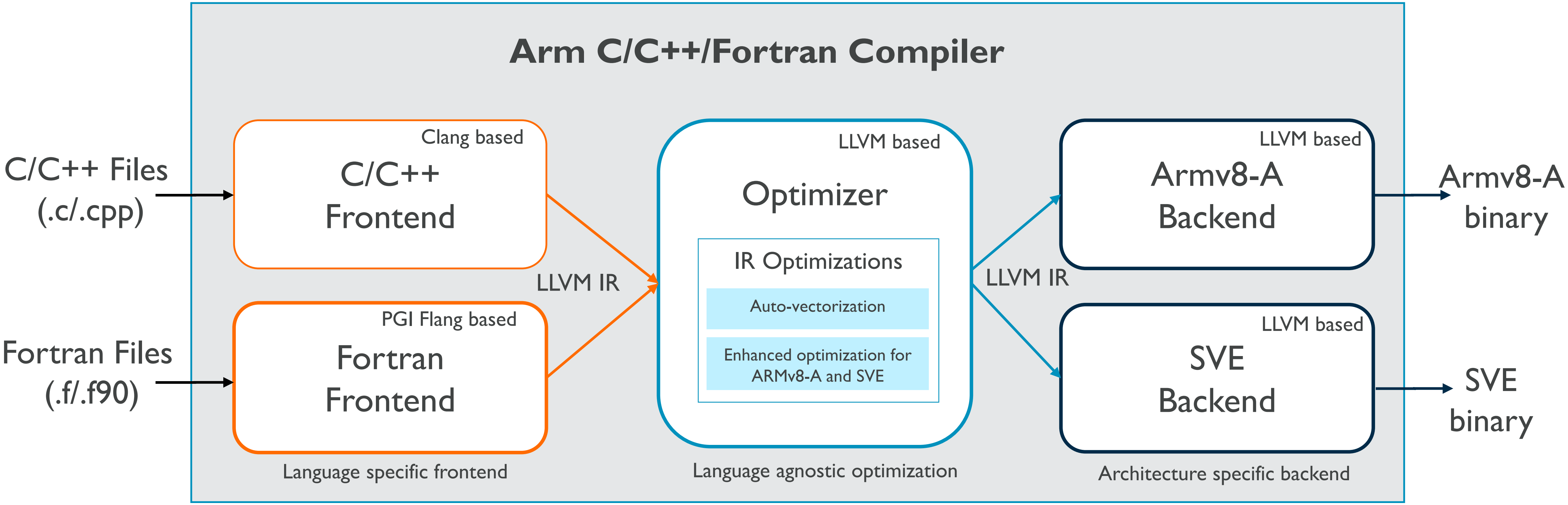
## Key benefits:

- Get the best performance out of Arm hardware
- Reduce the time to develop codes
- Improve productivity by speeding up applications

## and, importantly…

- Migrate from x86 to aarch64 faster with portable tools
- Forge/PR licences are a great justification to stay close to our end-users

# Arm Compiler – Building on LLVM, Clang and Flang projects

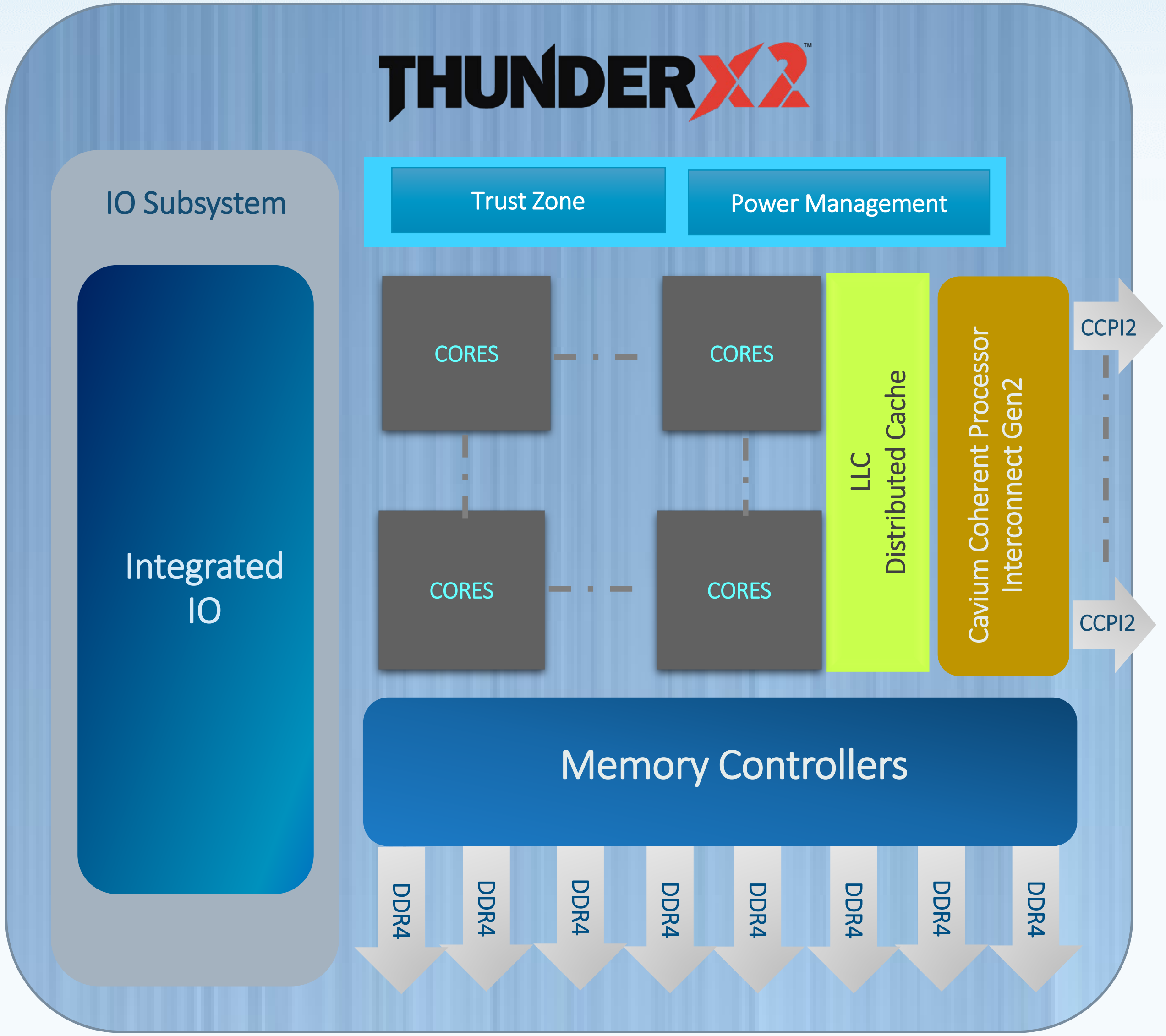


# Open Architecture implies other software tool options.

- GNU-gcc
  - including internal Arm developers
- LLVM-clang
  - including internal Arm developers
  - All enhancements to LLVM for the Arm HPC compilers are pushed upstream
    - Using PGI-flang as base for armflang
- Cray
  - Complete, HPC-optimized software stack including the Cray Linux® Environment and Cray Programming Environment. Cray's CCE compiler and programming environment are enhanced to achieve improved performance from the Cavium ThunderX2 processors.
  - I assume craypat is a functioning profiler on aarch64
- Fujitsu

# Hardware options

# Cavium CN99XX - 1st member of THUNDERX2 Family



- 24/28/32 Custom ARMv8 cores
- Fully Out-Of-Order (OOO) Execution
- 1S and 2S Configuration
- Up to 8 DDR4 Memory Controllers
- Up to 16 DIMMs per Socket
- Server Class RAS features
- Server class virtualization
- Integrated IOs
- Extensive Power Management

2nd gen ARM server SoC

Delivers 2-3X higher performance

CAVIUM

# Fujitsu Post-K specs

## Architecture Features

- Armv8.2-A (AArch64 only)
- SVE 512-bit wide SIMD
- 48 computing cores + 4 assistant cores*

*All the cores are identical

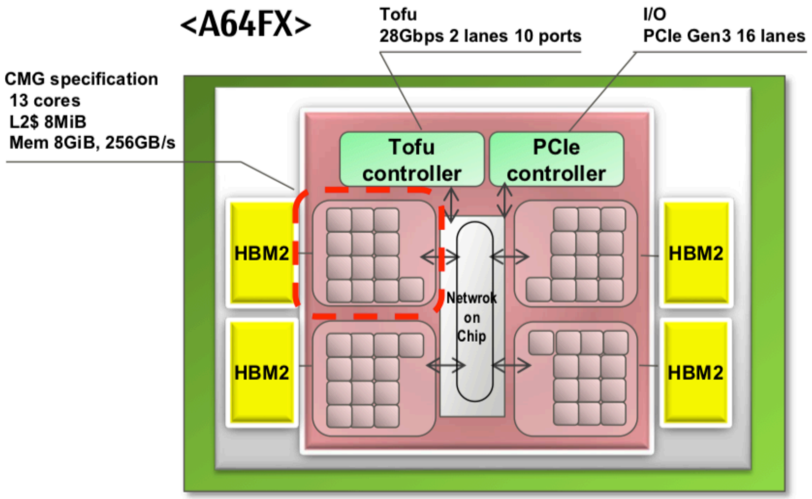
- HBM2 32GiB
- Tofu 6D Mesh/Torus
  28Gbps x 2 lanes x 10 ports
- PCIe Gen3 16 lanes

## 7nm FinFET

- 8,786M transistors
- 594 package signal pins

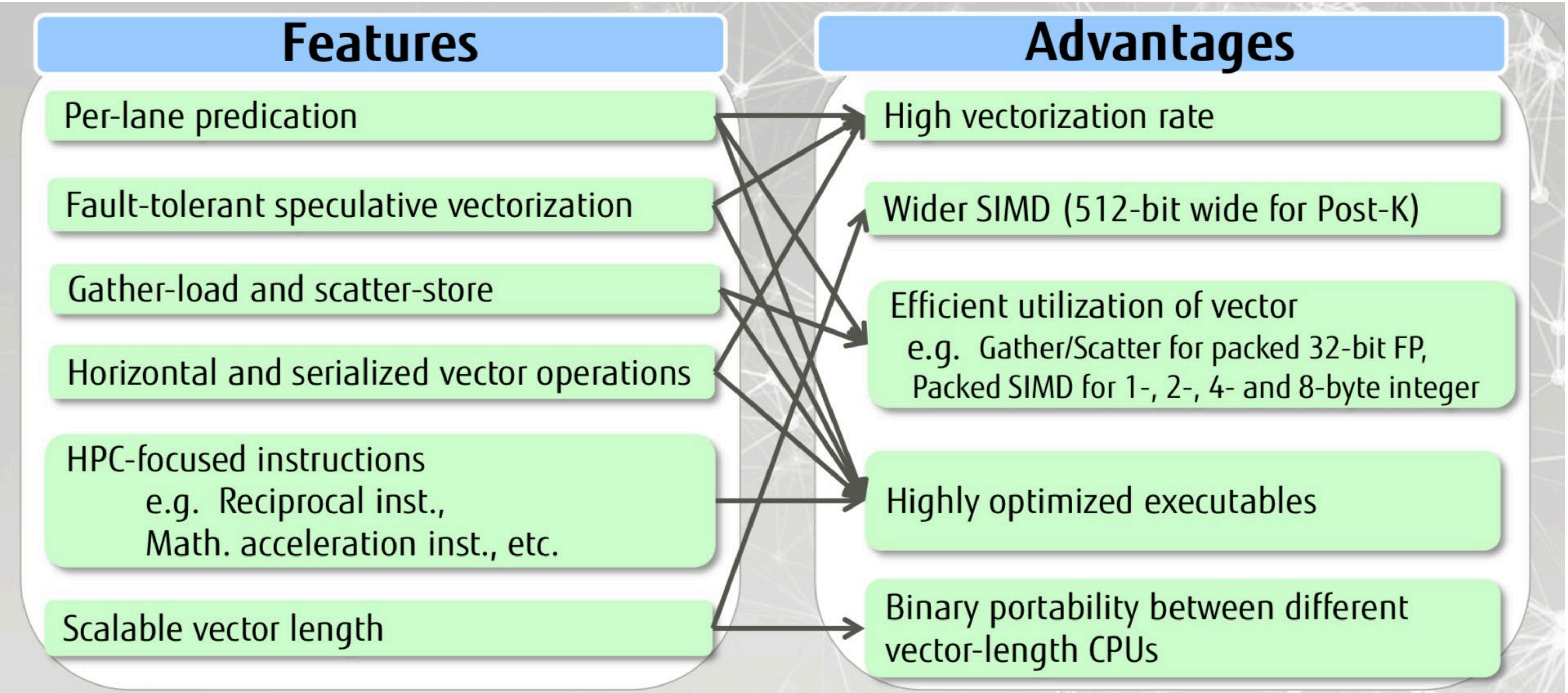
## Peak Performance (Efficiency)

- >2.7TFLOPS (>90%@DGEMM)
- Memory B/W 1024GB/s (>80%@Stream Triad)



| | A64FX (Post-K) | SPARC64 XIfx (PRIMEHPC FX100) |
|---|---|---|
| ISA (Base) | Armv8.2-A | SPARC-V9 |
| ISA (Extension) | SVE | HPC-ACE2 |
| Process Node | 7nm | 20nm |
| Peak Performance | >2.7TFLOPS | 1.1TFLOPS |
| SIMD | 512-bit | 256-bit |
| # of Cores | 48+4 | 32+2 |
| Memory | HBM2 | HMC |
| Memory Peak B/W | 1024GB/s | 240GB/s x2 (in/out) |

# Fujitsu Post-K attributes for HPC applications.



## Features

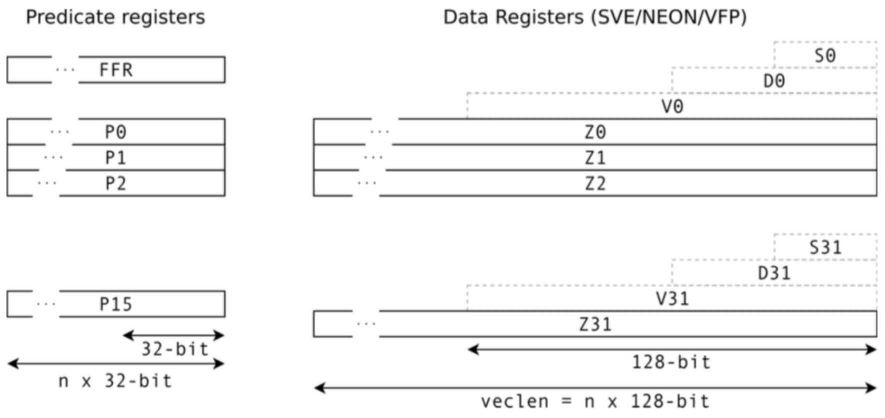
- Per-lane predication
- Fault-tolerant speculative vectorization
- Gather-load and scatter-store
- Horizontal and serialized vector operations
- HPC-focused instructions
  e.g. Reciprocal inst.,
  Math. acceleration inst., etc.
- Scalable vector length

## Advantages

- High vectorization rate
- Wider SIMD (512-bit wide for Post-K)
- Efficient utilization of vector
  e.g. Gather/Scatter for packed 32-bit FP,
  Packed SIMD for 1-, 2-, 4- and 8-byte integer
- Highly optimized executables
- Binary portability between different vector-length CPUs

# ArmV8 and HPC

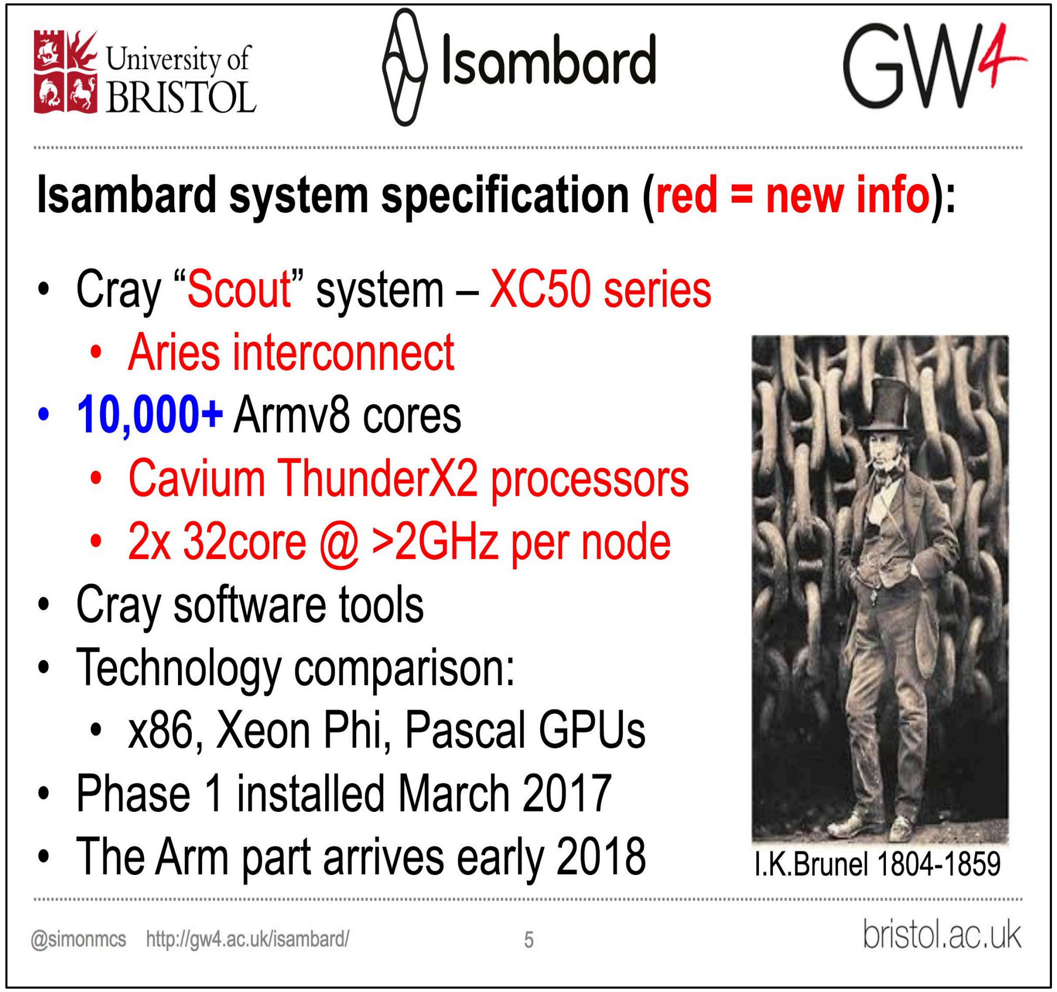
# Scalable Vector Extension



- Not an extension to NEON
- advanced loads & stores
- white-paper: https://developer.arm.com/-/media/developer/developers/hpc/white-papers/a-sneak-peek-into-sve-and-vla-programming.pdf?revision=c702475b-6325-41a2-b3d3-d9f244028841
  - Francesco Petrogalli
- Vector Length Agnostic (VLA) programming because of predication
  - can vectorize loops with control flow in the loop body
- Run SVE code on non–sve platforms with ARMIE: https://developer.arm.com/products/software-development-tools/hpc/arm-instruction-emulator
- SC’18 workshop

# Applications on Aarch64

# Bristol via GW4 is an active co-designer.



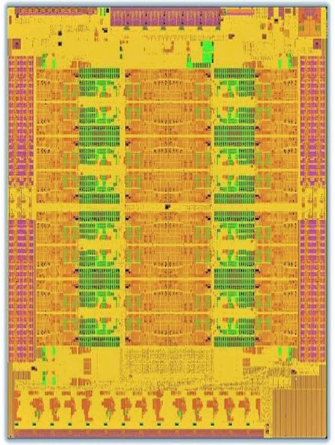
University of BRISTOL Isambard GW4

**Isambard system specification (red = new info):**

- Cray "Scout" system – XC50 series
  - Aries interconnect
- **10,000+** Armv8 cores
  - Cavium ThunderX2 processors
  - 2x 32core @ >2GHz per node
- Cray software tools
- Technology comparison:
  - x86, Xeon Phi, Pascal GPUs
- Phase 1 installed March 2017
- The Arm part arrives early 2018

I.K.Brunel 1804-1859

## Cavium ThunderX2, a seriously beefy CPU



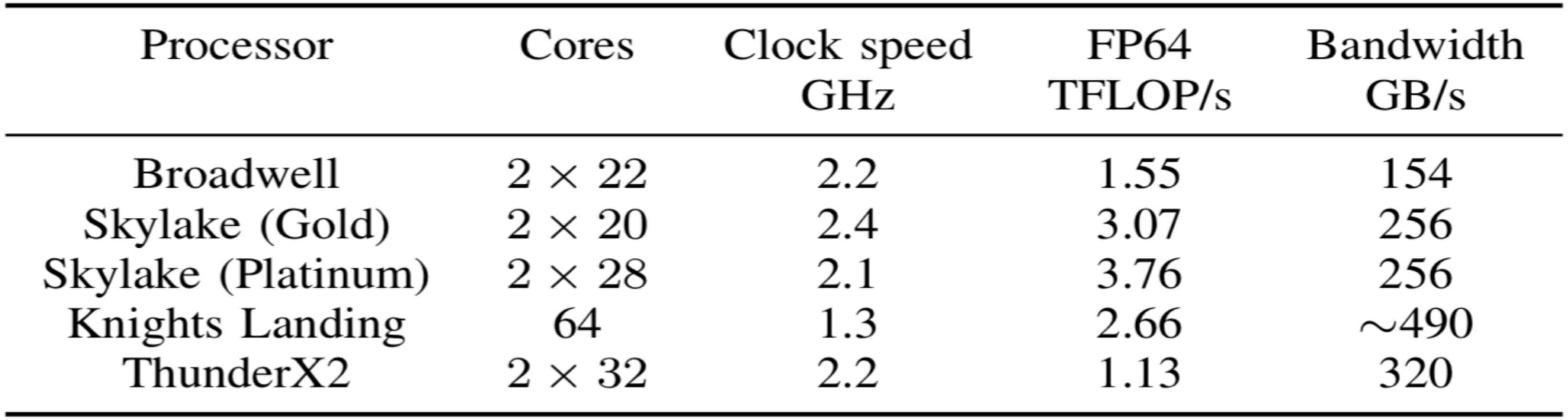
- 32 cores at up to 2.5GHz
- Each core is 4-way superscalar, Out-of-Order
- 32KB L1, 256KB L2 per core
- Shared 32MB L3
- Dual 128-bit wide NEON vectors
  - Compared to Skylake's 512-bit vectors, and Broadwell's 256-bit vectors
- 8 channels of 2666MHz DDR4
  - Compared to 6 channels on Skylake, 4 channels on Broadwell
  - AMD's EPYC also has 8 channels

University of BRISTOL

http://gw4.ac.uk/isambard/

GW4

# The orange is different from the apples.



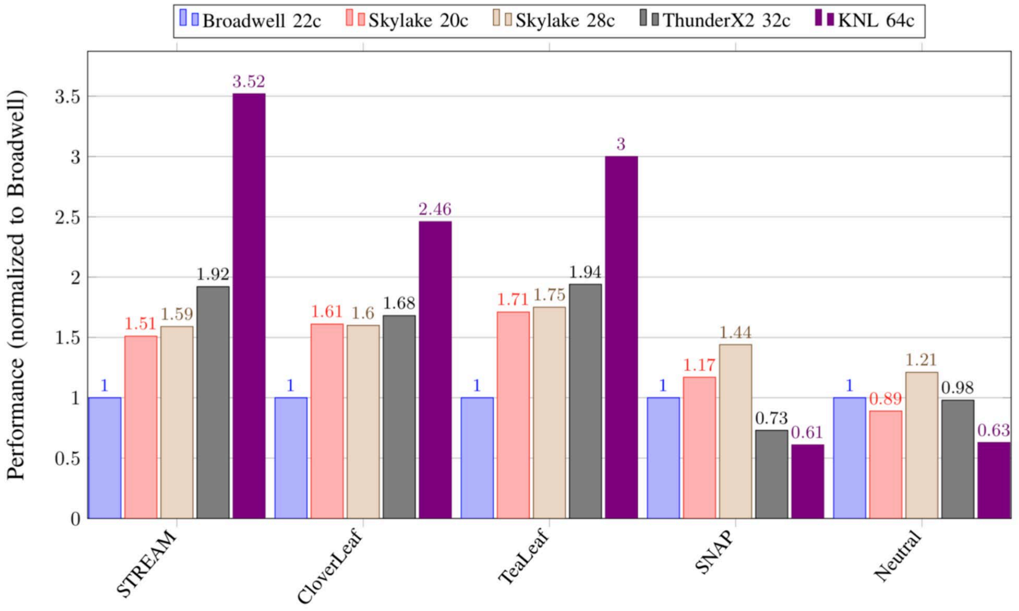
| Processor | Cores | Clock speed<br>GHz | FP64<br>TFLOP/s | Bandwidth<br>GB/s |
|---|---|---|---|---|
| Broadwell | 2 × 22 | 2.2 | 1.55 | 154 |
| Skylake (Gold) | 2 × 20 | 2.4 | 3.07 | 256 |
| Skylake (Platinum) | 2 × 28 | 2.1 | 3.76 | 256 |
| Knights Landing | 64 | 1.3 | 2.66 | ∼490 |
| ThunderX2 | 2 × 32 | 2.2 | 1.13 | 320 |

**BDW 22c** Intel Broadwell E5-2699 v4, **$4,115** each (near top-bin)
**SKL 20c** Intel Skylake Gold 6148, **$3,078** each
**SKL 28c** Intel Skylake Platinum 8176, **$8,719** each (near top-bin)
**TX2 32c** Cavium ThunderX2, **$1,795 each** (near top-bin)

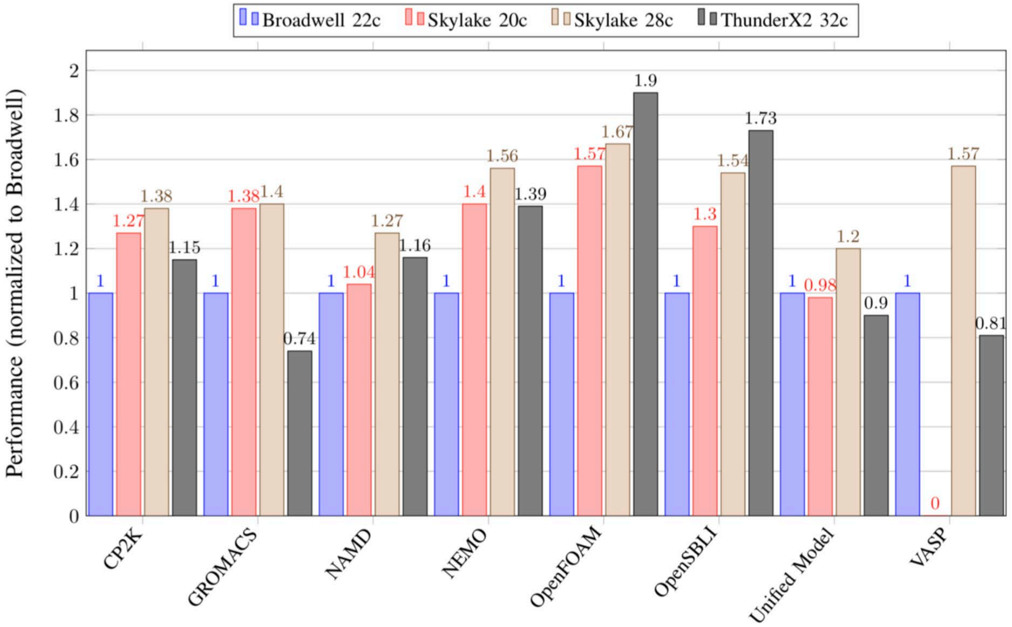
http://gw4.ac.uk/isambard/

# Memory bound apps do perform better on TX2.

## Performance on mini-apps



## Performance on heavily used applications from Archer

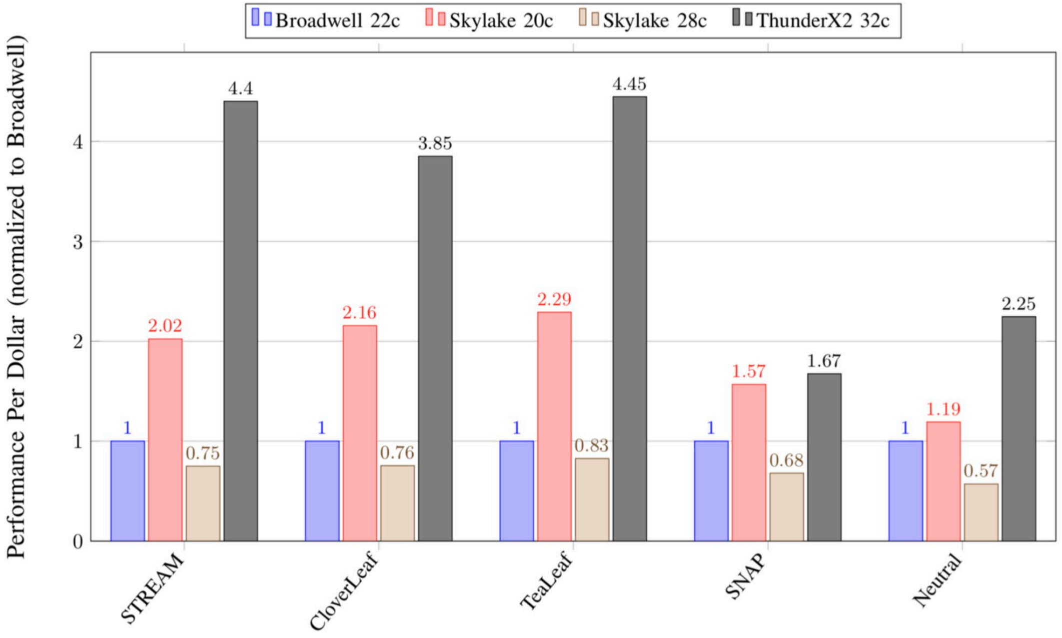


**Comparative Benchmarking of the First Generation of HPC-Optimised Arm Processors on Isambard**

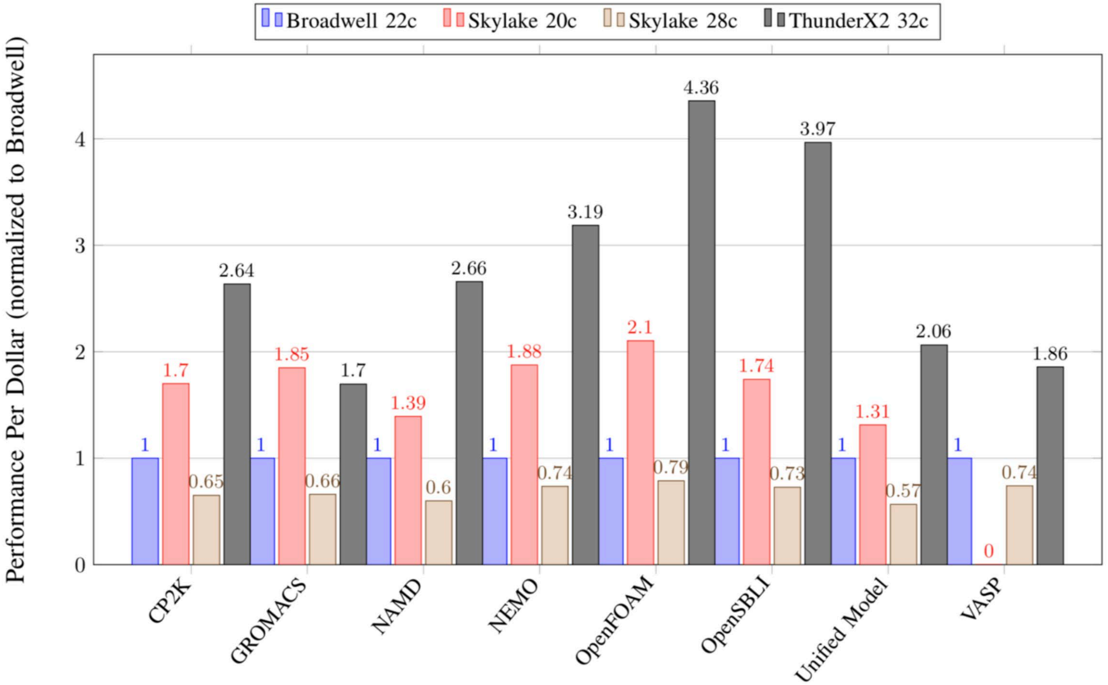
S. McIntosh-Smith, J. Price, T. Deakin and A. Poenaru, CUG 2018, Stockholm

# What is the cost for your performance?

## Performance per Dollar: mini-apps

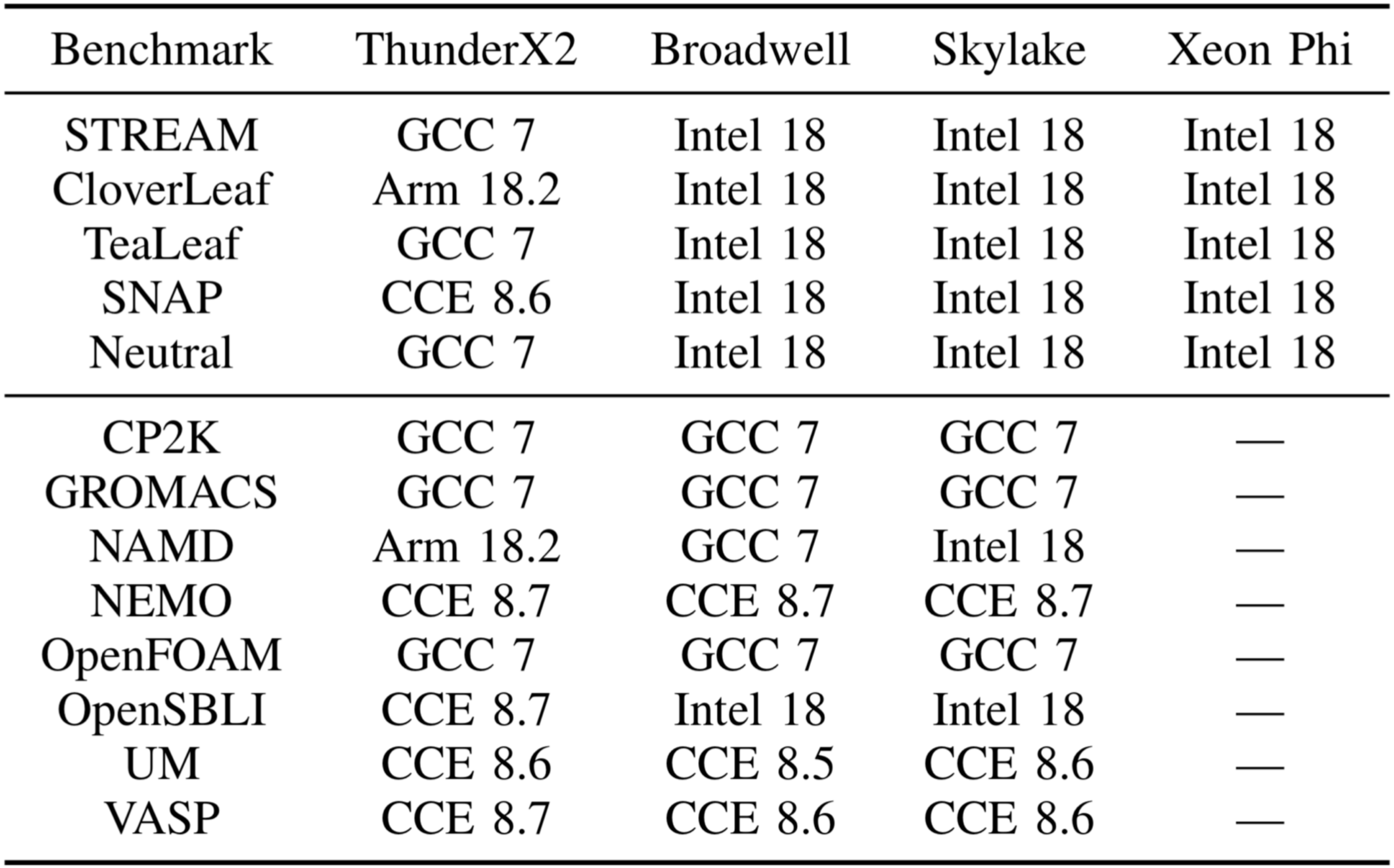


## Performance per Dollar: applications



This is highly subjective to the market and procurement deals.

# No single compiler is the best.



| Benchmark | ThunderX2 | Broadwell | Skylake | Xeon Phi |
|---|---|---|---|---|
| STREAM | GCC 7 | Intel 18 | Intel 18 | Intel 18 |
| CloverLeaf | Arm 18.2 | Intel 18 | Intel 18 | Intel 18 |
| TeaLeaf | GCC 7 | Intel 18 | Intel 18 | Intel 18 |
| SNAP | CCE 8.6 | Intel 18 | Intel 18 | Intel 18 |
| Neutral | GCC 7 | Intel 18 | Intel 18 | Intel 18 |
| CP2K | GCC 7 | GCC 7 | GCC 7 | — |
| GROMACS | GCC 7 | GCC 7 | GCC 7 | — |
| NAMD | Arm 18.2 | GCC 7 | Intel 18 | — |
| NEMO | CCE 8.7 | CCE 8.7 | CCE 8.7 | — |
| OpenFOAM | GCC 7 | GCC 7 | GCC 7 | — |
| OpenSBLI | CCE 8.7 | Intel 18 | Intel 18 | — |
| UM | CCE 8.6 | CCE 8.5 | CCE 8.6 | — |
| VASP | CCE 8.7 | CCE 8.6 | CCE 8.6 | — |

# The HPC community has options to influence Arm

- Arm has made the effort to <help> port many applications:
  - https://gitlab.com/arm-hpc/packages/wikis/categories/allPackages
  - Please contribute and enhance ported packages during your campaigns
  - support-hpc-sw@arm.com  with concerns
  - https://spack.io
    - HPC deployment package manager
    - patch in for IP review for armhpc compiler
    - ?standardization of benchmarks?
  - SVE (starting with 512 bit vector units) will bring more performance to the table
    - compiler implementations will always be advancing

# Arm help

# HPC Infrastructure Tools Group

Worldwide HPC knowledge experts in cross-platform enablement and performance optimization

## Teams:

- Professional Services
- Commercial and open-source SW engineering
- HPC Tools Development, training and support

## Offerings:

### HPC application modernization, porting, and optimization:

- Identify and resolve scalability bottlenecks.
- Optimize application data movement and vectorization.

### Performance analysis tools and techniques:

- Deploy and support software tools from Arm and the Arm community.
- Documentation, training, tutorials, and workshops.

### Knowledge transfer and Arm adoption:

- Publications, presentations, and community participation.

Thank You!

Danke!

Merci!

谢谢!

ありがとう!

Gracias!

Kiitos!

감사합니다

धन्यवाद

arm