

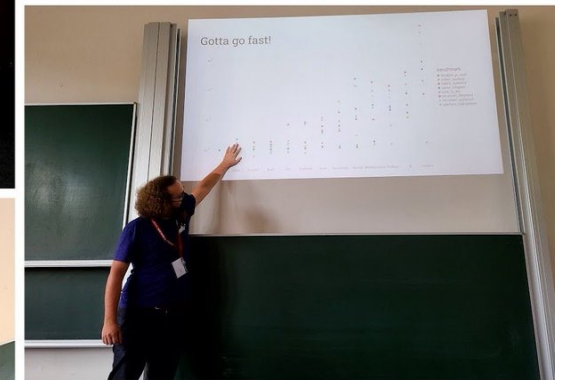
The background of the slide is a photograph of a mountain range, likely the Blue Mountains in Australia, with a prominent peak on the left. The mountains are layered, creating a sense of depth. A solid black rectangular box is positioned in the lower half of the image, containing the event title and date in white text.

ARM HPC USER GROUP SC22 SYMPOSIUM

14 NOVEMBER 2022

AHUG 2022 RECAP

- IWAHPCE 2022
- ISC22 Workshop and BoF
- EAHPC22
- SC22 Symposium
- SC22 BoF
- IWAHPCE 2023



← → ↻ arm-hpc-user-group.github.io/iwahpce-2023/

International Workshop on Arm-based HPC: Practice & Experience

An HPCAsia 2022 Workshop

[View On GitHub](#)

This project is maintained by [arm-hpc-user-group](#)

International Workshop on Arm-based HPC: Practice and Experience (IWAHPCE-2023)

to be held in conjunction with The International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2023), Singapore, Feb 27 - March 2, 2023

Workshop Overview

This workshop aims to provide the opportunity to share the practice and experience of high-performance computing systems using the Arm architecture and their performance and applications. The last few years have seen an explosion of 64-bit Arm-based processors targeted toward server and infrastructure workloads, often specializing in a specific domain such as HPC, cloud, and machine learning. Fujitsu's A64FX and Marvell's ThunderX2 have been used in several large-scale HPC systems, and Amazon's Graviton2 has been adopted by Amazon EC2. Moreover, Amazon's Graviton3, NVIDIA Grace CPU Superchip, and SiPearl's Rhea system-on-chip are recently announced or become accessible. Sharing the practice and experiences using these Arm-based processors will contribute to advancing high-performance computing technology for newly designed systems using these new Arm-based processors.

SC 2022 BIRDS OF A FEATHER

Arm Diversity Unified: Standardization in Hardware and Software

**November 15th, 5:15-6:45 PM CST
C141-143-149**



AHUG ELECTIONS – 2022

- Thank you to our previous Board of Directors who went above and beyond!
 - Jeffrey Young, Steve Poole, Mitsuhsa Sato, Michèle Weiland, Valerio Schiavoni, Christelle Piechurski
- The Board of Directors rotate on two year terms
 - The next board will serve for 2023-2024 with elections in Fall of 2024
- 40 voters in the AHUG 2022 Election for BoD
 - Full results will be shared via the mailing list and Slack group

AHUG BOARD 2023-2024

- Managing Director - Simon McIntosh-Smith
- Associate Directors - Eva Siegmann, Filippo Spiga, Miwako Tsuji,
- Secretary - Eric Boyer
- Treasurer – Valerio Schiavoni




AHUG BYLAWS

- We have new bylaws!
 - But we've elected to let the new BoD approve them fully and share them.
- Key pieces of the new bylaws
 - Who can be a member?
 - Any individual can be a member. Voting members will have attended an AHUG event in the past 12-18 months.
 - AHUG is envisioned to be a sponsor-driven organization so no individual member fees currently.
 - Industry Steering Committee
 - Each company can appoint a member to the committee
 - The steering committee will work with the BoD to drive the agenda and direction of AHUG
 - Formalization of the election process and process for replacing BoD and steering committee members.
 - Code of Conduct for AHUG events and interactions

AHUG RESOURCES

Don't forget to take advantage of the many AHUG resources!

- Github organization - <https://github.com/arm-hpc-user-group>
- YouTube
- Slack
- LinkedIn Group
- Mailing List - <https://arm-hpc.groups.io/g/ahug/>



Arm HPC User Group

[Overview](#) [Repositories 13](#) [Projects](#) [Packages](#) [Teams 2](#) [People 7](#) [Settings](#)

README.md

Arm HPC User Group Github Organization

The Arm HPC User Group – or AHUG, is a user-led, not-for-profit organization intent on promoting the latest Arm-based silicon, systems and platforms for the High Performance and Technical Computing community. The organization is open to all members, and our focus is on supporting community engagement and education via workshops, tutorials, and hackathons. Please see our [main website](#) for more information on the organization. This Github presence is meant to provide access to materials provided at recent events.

Recent AHUG Events

- [IWAHPCE 2023](#) - a proceedings-based workshop focused on Arm research and work related to Arm supercomputers
- [Embracing Arm for HPC 2022](#) - a proceedings-based workshop on the latest Arm HPC research
- [ISC AHUG 2022 Workshop](#) - a non-proceedings workshop focused on community efforts and updates in the Arm HPC space.

How to Get Engaged with AHUG

- Join one of our events! See [this page](#) for more details on official AHUG events.
- Join the [AHUG Slack](#) and/or [mailing list](#).

COMMENTS FROM THE NEW MANAGING DIRECTOR



LOTS OF PROGRESS IN THE ARM HPC COMMUNITY!

Hardware:

- Fugaku
- AWS Graviton 2 & 3
- Apple M1 / M2...
- In 2023: NVIDIA Grace

Software:

- More and more packages ported
- Compiler maturity for SVE etc
- NVIDIA bringing their AI/ML stack to Grace
- Spack on AWS
- R...

ARM COMMUNITY: R ON AARCH64



- What is R?
 - Dynamically typed programming language for statistical computing
 - Rich package ecosystem
- CRAN (Comprehensive R Archive Network)
 - ~19,100 packages, R and Native (C/C++, Fortran) packages, 101GB of source tarballs
- Bioconductor (molecular dynamics focused R packages)
 - ~2,183 packages R and Native (C/C++, Fortran) packages, 6.1GB of source tarballs
- Highly coupled
 - Many packages require: X11, BLAS, LaTeX, Pandoc (Pandoc then requires Haskell...)
 - Some packages have ~2,000 recursive dependencies (direct dependencies + their dependencies, recursively)
 - Some CRAN needs Bioconductor packages, vice versa.
- Wide range of package languages: R wraps code written in:
 - Fortran
 - C/C++ (Extensive use of OpenMP, BLAS, and even MPI)
 - Rust
 - Java (mostly bindings)
 - Objective Pascal (Oberon derivative, e.g. OpenBUGS)

R ON AARCH64: CRAN BUILD CHECKS

- R package install process (simplified)
 - Install
 1. Untar
 2. Validate R code
 3. Compile foreign (non-R) code
 4. Validate package loading
 - Check
 1. Run unit/integration tests
 2. Generate vignettes (LaTeX/Markdown => PDF)
- CRAN packages are installed and checked everyday
 - Changes to a single package affect multiple dependencies
 - Installs and check 19,100 (+ some of the 2,183 from Bioconductor) packages everyday
 - CRAN cover x86 Linux, MacOS, Windows, and M1 Mac
 - Multiple Linux distros
 - **AArch64 Linux is missing**

Flavor	OK	NOTE	WARN	ERROR	FAIL	Total	
r-devel-linux-x86_64-debian-clang	14644	3720	329	122	2	18817	Details
r-devel-linux-x86_64-debian-gcc	14749	3726	242	105	2	18824	Details
r-devel-linux-x86_64-fedora-clang	12327	6042	328	105	0	18802	Details
r-devel-linux-x86_64-fedora-gcc	13124	5358	241	87	0	18810	Details
r-devel-windows-x86_64	14685	3982	23	102	18	18810	Details
r-patched-linux-x86_64	15476	3214	16	117	1	18824	Details
r-release-linux-x86_64	15515	3207	15	84	3	18824	Details
r-release-macos-arm64	13085	5628	12	97	0	18822	Details
r-release-macos-x86_64	13138	5562	15	104	0	18819	Details
r-release-windows-x86_64	15237	3453	16	88	16	18810	Details
r-oldrel-macos-arm64	13018	5669	30	81	2	18800	Details
r-oldrel-macos-x86_64	13138	5526	38	86	0	18788	Details
r-oldrel-windows-ix86+x86_64	14763	3606	150	208	14	18741	Details

https://cran.r-project.org/web/checks/check_summary.html

R ON AARCH64: LATEST RESULTS

- Bristol has repurposed two early Marvell ThunderX2 nodes as a new build system
- Similar CRAN install/check script, but uses Docker-compose
- Results
 - AArch64 (TX2 dual socket = ~2.5h install + ~6h check on one node only)
 - Failed install: 8, (16 packages depend on these and did not install)
 - X86 (Xeon Gold 6126 single socket, ~23h total check + install)
 - Failed install: 23 (203 packages depend on these and did not install)
- Current analysis
 - Most packages work fine on AArch64 after fixing a few major blockers: MPI, QuantLib, X11, etc.
 - Three major categories for failed install/checks on AArch64
 - Package has old libtool platform detection script from early 2000; AArch64 didn't exist.
 - Easy fix, send PR, wait.
 - Package has algorithms with hand coded SSE/AVX, popular in bioinformatics (e.g Smith Waterman).
 - Implement slow path, send PR, wait.
 - Proprietary binary projects: Bloomberg APIs, Oracle packages, Dead languages (Objective Pascal)
 - No solution unless vendors come in, around 5 ~10 of these.

<https://github.com/UoB-HPC/r-run-check>