C++ Standardization and Evolution for Computational Astrophysics

This presentation constitutes a report on my C++ standardization activities that has been funded in part by the Action Spécifique Numérique over the last few years. I will review the standardization work that has taken place in 2024 and 2025. I will discuss existing and upcoming C++ features that can greatly benefit computational astrophysics. Topics will include: concepts, parallel executors, SIMD, compile-time programming, linear algebra and many other smaller features. I will also give details about the ongoing work on reflection which is expected to be a transforming capability of C++26. For the high-performance community, it will, for example, ease the MPI serialization of data and allow to automate Arrays-of-Structures to Structures-of-Arrays transformations on generic containers. I will also shortly discuss the current state of the C++ ecosystem for HPC, including Kokkos, SYCL, Vulkan, and EVE.

Note: As a note this presentation is tightly coupled with the one of Tetautahi MAAMAATUAIAHUTAPU which will illustrate many of the features discussed in this talk. Ideally these two talks make a set together, this one being a general introduction, and the one on integral transforms being a concrete illustration. If a choice between the two has to be made, it would probably be best to keep the one on integral transforms.