Contribution ID: **2** Type: **Talk 25'**

Enabling Julia code to run at scale with artefact caching

Monday 30 September 2024 15:00 (30 minutes)

The Julia programming language has evolved into a mature tool for scientific computing over the past decade, offering high-level capabilities with just-in-time (JIT) compilation and efficient garbage collection. Its performance, comparable to that of C/C++, makes Julia an attractive option for the high-energy physics community. However, Julia's use of precompiled files to achieve this performance introduces significant startup delays on the first program execution ("time to first plot"), which would be a severe drawback in distributed systems where each node would have to compile dependencies locally. This project proposes a workflow to obtain a ready-to-use directory with all necessary precompiled files for selected applications, while also leveraging the shared CernVM-FS (CVMFS) file system to share files across nodes in a distributed setup. We developed and tested a framework for automating the publication of precompiled Julia files to CVMFS and evaluated its impact on performance using two sample applications: the Julia Jet Reconstruction package and the Geant4 wrapper package. Our findings demonstrate that caching precompiled files that are stored in CVMFS significantly reduces startup times, with reductions of up to 97% for the Geant4 package. Furthermore, we examined the effects of cross-compilation for various microarchitectures and found that nodes benefit from shared cache files without notable performance degradation due to microarchitecture differences.

Author: Mr AGUERO VERA, Elvis Alexander (Brown University)

Co-authors: MATO VILA, Pere (CERN); GRAEME, Stewart

Presenter: Mr AGUERO VERA, Elvis Alexander (Brown University)

Session Classification: Talks