

Expressing Parallelism in ROOT

Monday, 10 October 2016 14:45 (15 minutes)

The need for processing the ever-increasing amount of data generated by the LHC experiments in a more efficient way has motivated ROOT to further develop its support for parallelism. Such support is being tackled both for shared-memory and distributed-memory environments.

The incarnations of the aforementioned parallelism are multi-threading, multi-processing and cluster-wide executions. In the area of multi-threading, we discuss the new implicit parallelism and related interfaces, as well as the new building blocks to safely operate with ROOT objects in a multi-threaded environment. Regarding multi-processing, we review the new MultiProc framework, comparing it with similar tools (e.g. multiprocessing module in Python). Finally, as an alternative to PROOF for cluster-wide executions, we introduce the efforts on integrating ROOT with state-of-the-art distributed data processing technologies like Spark, both in terms of programming model and runtime design (with EOS as one of the main components).

For all the levels of parallelism, we discuss, based on real-life examples and measurements, how our proposals can increase the productivity of scientists.

Tertiary Keyword (Optional)

High performance computing

Secondary Keyword (Optional)

Analysis tools and techniques

Primary Keyword (Mandatory)

Parallelization

Primary authors: PIPARO, Danilo (CERN); TEJEDOR SAAVEDRA, Enric (CERN)

Co-authors: GUIRAUD, Enrico (Università degli Studi e INFN Milano (IT)); GANIS, Gerardo (CERN); MONETA, Lorenzo (CERN); MATO VILA, Pere (CERN); CANAL, Philippe (Fermi National Accelerator Lab. (US)); VALLS PLA, Xavier (University Jaume I (ES))

Presenter: TEJEDOR SAAVEDRA, Enric (CERN)

Session Classification: Track 5: Software Development

Track Classification: Track 5: Software Development