



Contribution ID: 386

Type: **Oral**

## Migrating large codebases to C++ Modules

*Wednesday 13 March 2019 19:00 (20 minutes)*

ROOT has several features which interact with libraries and require implicit header inclusion. This can be triggered by reading or writing data on disk, or user actions at the prompt. Often, the headers are immutable and reparsing is redundant. C++ Modules are designed to minimize the reparsing of the same header content by providing an efficient on-disk representation of C++ Code. ROOT has released a C++ Modules-aware technology preview which intends to become the default for the next release.

In this contribution, we would like to summarize our ROOT experience migrating to C++ modules the codebase of ROOT. We outline the challenges for migration of the CMS software stack to use C++ modules, including integration of modules support in the build system while providing better functionality and correctness. We also give an insight of the continuous process of the improving performance bottlenecks for C++ modules and also evaluate the performance benefits that experiments are expected to achieve.

**Primary authors:** SHADURA, Oksana (National Academy of Sciences of Ukraine (UA)); VASILEV, Vasil Georgiev (Princeton University (US)); TAKAHASHI, Yuka (University of Tokyo)

**Presenter:** TAKAHASHI, Yuka (University of Tokyo)

**Session Classification:** Track 1: Computing Technology for Physics Research

**Track Classification:** Track 1: Computing Technology for Physics Research