Sample Project: Development of a free and open source C++ static code analysis tool

Code	EP5786
Programme	FCT
Department	EP
Responsible	16919 - Dr. John Harvey
Created by	110252 - Mr. Joschka Philip Lingemann
Updated by	96245 - Mr. Vasco Miguel Chibante Barroso
Date Created	12-AUG-16
Date updated	17-AUG-16

Title

Development of a free and open source C++ static code analysis tool

Description

Typical scientific and industrial software projects have many developers with different backgrounds. Testing and code quality checks spot potential problems early on and increase the productivity of such teams. Ultimately such checks guarantee a high quality software product. One ingredient to these quality checks is a static analysis of changes to the code base, which aims to identify defects even before compiling the project.

The open source Static Analysis Suite (SAS) is an alternative to existing expensive commercial solutions. SAS is based on the widely adopted Clang compiler toolkit and in particular its static analyser. SAS not only allows traditional static analysis but also to implement checks for compliance with coding conventions. SAS is currently used by experiments in High Energy Physics but could also find application outside the field.

The trainee will start by extending the already existing set of checkers in SAS. These new checkers could help to detect common problems in multithreading, find potential performance limiters or even security vulnerabilities. An important part of a static code analyser is a good interface for easy inclusion into continuous integration solutions. The trainee will also be responsible for this integration by writing the needed tools for commonly used solutions, such as Jenkins and Travis. Once these tools are ready, existing projects at CERN could be approached as potential new users. The trainee could assist in integrating SAS into their build procedures, giving the trainee insights into different communities and applications in HEP. This exposure would also likely lead to further extensions of the existing tests.

Skills

Information Technologies: Using software development tools (e.g. Git, Jira, Trac). Programming Languages: C++, Python <p class="p1">
 Experience with Continuous Integration solutions is a plus.</p>

Disciplines

Information Technologies