CerTran: Effortless, continuous C/C++ compiler qualification

Parma/Amsterdam, March 9, 2023

Compiler qualification is one of the requirements for using C/C++ in the development of critical systems. Qualification by validation is by far the most convenient methodology among those that are allowed by functional safety standards such as ISO 26262. And it is not rocket science: with SuperTest you acquire a validation test-suite containing thousands of validated tests, configure it for your particular use cases, run its test automation machinery, and then see whether some tests have failed and what to do about them. The first time you do it, configuration will take a couple of days to a week. If you do that often and become fluent in it you can produce a new configuration in a few hours. Actually, the hardest part of the configuration is coming up with the list of "use cases" which you need to qualify: simplifying a bit, a "use case" is defined by a particular version of a particular compiler and the precise set of compiler options you give to it.

Obtaining that set of options along with their defaults and capturing it in the validation test-suite configuration requires a deep knowledge of the compiler and of the build system(s) used by the project: a mistake committed at this stage jeopardizes the entire qualification effort.

Manual configuration of the validation test-suite, besides being onerous and error-prone, does not match current industry trends:

1. Systems are composed by an increasing number of components, which often come from different sources and have their own build procedures and use the compilers in different ways. The number of components of a modern ADAS system may be more than a hundred, and it may have dozens of different compiler use cases (e.g., floating-point-intensive components will use peculiar combinations of options to control floating-point accuracy and performance, other component may use completely different sets of options in order to deal effectively with their own trade-offs).

2. A growing number of organizations are embracing continuous integration, for different reasons, mainly because they recognize the value of automation and repeatability, and "shift-left" of all the verification activities. And compiler qualification is one such activity: if there is a problem in your compiler use case, you want to know earlier rather than later; if for some reason you need to tweak your compiler use case (e.g., to squeeze some extra performance or to work around a bug), you want to immediately know whether you are bumping into a/another compiler defect.

CerTran is a joint solution developed by **BUGSENG** and **Solid Sands**.

It leverages one of the functionalities ECLAIR possesses since ever: interception of the invocation of all the translation toolchain components (the compiler, to start with) independently from the build procedure, and the full "understanding" of all the options passed (in whichever way) to the compiler and how each option influences the compiler behavior.

For more information visit
https://www.bugseng.com/eclair/certran

This amounts to precisely, automatically and reliably recognize all use cases of the translation toolchain for each given project. CerTran automatically produces one SuperTest configuration for each detected use case and executes the SuperTest automation machinery to qualify the compiler for all of them.

The full automation of the entire procedure allows running CerTran into a CI/CD system such as GitLab and Jenkins, thereby making *continuous compiler qualification* a reality.

Whether or not you intend to use CI/CD systems, CerTran makes the task of qualifying multiple toolchain use cases a matter of minutes instead of hours or days.

Automated translation toolchain validation:
a CI-ready solution for excellence and efficiency in development and qualification

cer (T) ran

bugSeng

BUGSENG is a leading provider of solutions and services for software verification.

Implementing state of the art technology, BUGSENG designed the ECLAIR Software Verification Platform®. ECLAIR is a powerful platform for the automatic analysis and verification of C and C++ programs.
The hardest part of the configuration, namely the adaptation to the compiler toolchain and the particular language dialect(s) used in the project, is **fully automatic**.
ECLAIR grants uniform user experience whether it is used from its **GUI,** from the **command line**, within **IDEs** and in **CI/CD systems** such as Jenkins, GitLab and GitHub.

ECLAIR is **certified** by TÜV SÜD Rail GmbH for use in safety-critical developments according to the main functional safety standards, up to the highest levels of criticality.

BUGSENG consulting services help industry leaders improving their development processes and complying with functional-safety standards.
BUGSENG is also a renowned resource for advanced professional training.
Visit: **http://bugseng.com**.

Solid Sands is the leading provider of compiler and library testing and qualification technology in North-America, Europe and Asia.

Founded in 2014, Solid Sands is headquartered in Amsterdam, The Netherlands. With partners all over the world we serve both leading innovative companies in the semiconductor, IP and security industries as well as safety-critical companies in automotive, robotics, railway and medical.

Our SuperTest Compiler Test and Validation Suite provides a complete validation environment which enables customers to achieve the software quality level demanded by the ISO language and functional safety standards.

Meanwhile, our SuperGuard C Library Safety Qualification Suite is a requirements-based test suite for the C standard library with full traceability between the requirements derived from the ISO C language definition and the individual library tests.
Visit: **https://solidsands.com**

no shortcuts,
no compromises,
no excuses:
software verification
done right