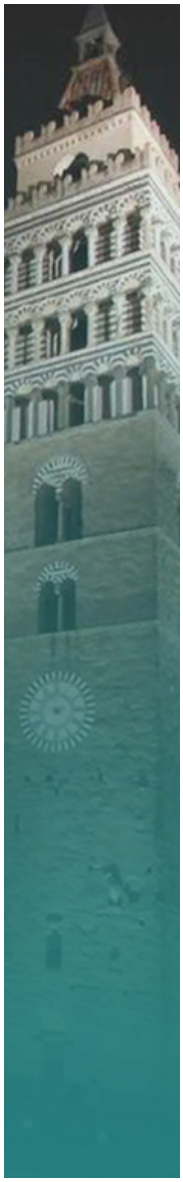


Tutorials

Mini-tutorials will be organised during the morning of Tuesday, 14 November, 2017 – before the main conference programme starts that afternoon.

Four tutorials are available, and delegates should use the registration page to indicate that they wish to attend none, one or two of these tutorials.

- Tutorial 1. AdaCore technologies for CENELEC EN 50128 2011 (Presented by Eric Perlade from AdaCore).
- Tutorial 2. Low cost high-integrity platform (Presented by Thierry Lecomte and Patrick Péronne from ClearSy).
- Tutorial 3. Model-Checking a Railway Interlocking System (Presented by Nicolas Breton from Systerel).
- Tutorial 4. REVEAL: Requirements elicitation, documentation and management approach (Presented by Andrew Hawthorn from Altran).

A full timetable for the tutorials will appear soon, but please note that the running order is:
Room 1: Tutorial 1; coffee break; Tutorial 4
Room 2: Tutorial 2; coffee break; Tutorial 3
Thus the combination of Tutorials 1 & 2 is not possible, and the same is true for Tutorials 3 & 4. All other selections are permitted. Please specify your selection using the registration page.

AdaCore CLEARSY SYSTEMS ENGINEERING systerel Safe real-time solutions ALTRAN

Tutorial 1. AdaCore technologies for CENELEC EN 50128 2011

Presented by Eric Perlade from AdaCore.

This workshop will be presenting the usage of AdaCore's technology in conjunction with the CENELEC EN 50128:2011 standard. It will describe where the technology fits the best and how it can be used to meet various requirements of the standard.

AdaCore's technology revolves around programming activities, as well as the closely-related design and verification activities. This is the bottom of the V cycle as defined by section 5.3 in EN 50128. It is based on the features of the Ada language (highly recommended by table A.15), in particular its 2012 revision, which adds some significant capabilities in terms of specification and verification.

The following tools and technology will then be presented and demonstrated:

- Ada 2012 language
- SPARK 2014 language and verification toolset performing formal proof and verification
- GNAT compiler
- CodePeer - static analysis tool that identifies potential run-time errors

- GNATmetric - metric computation tool
- GNATcheck - coding standard checker
- GNATdashboard - metric integration and management platform
- GNATtest - testing framework generator
- GNATemulator - processor emulator
- GNATcoverage - structural code coverage checker

Those tools will be presented in relation to the design, implementation, testing andintegration phases.

In addition, the contributions of the technologies to the Software Quality AssurancePlan will also be shown using tables from annex A - Criteria for the Selection of Techniques and Measures.

The tutorial using some example will be a quick usage guide covering aspects likeboundary value analysis, control flow analysis, data flow analysis, defensive programming, impact analysis and formal methods as referenced in annex D.

Tutorial 2. Low cost high-integrity platform

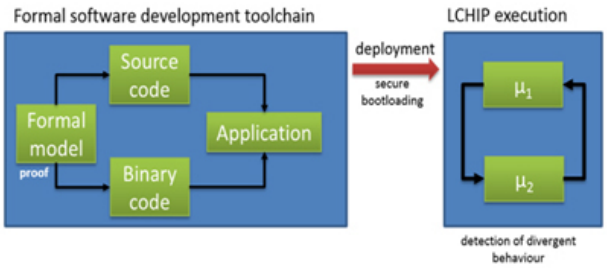
Presented by Thierry Lecomte and Patrick Péronne from ClearSy.

Low cost high-integrity platform (LCHIP) is a new technology aimed at revolutionizing the development of lightweight safety critical applications, up to SIL4. LCHIP building blocks are already embedded in certified railway applications (platform screen-doors controllers of Stockholm Citybanan and São Paulo L15 metros) and are expected to significantly lower the development and production costs of such systems.

LCHIP is based on two concepts:

- A unique formal model (B language) of a function, to obtain a safe application embedding two redundant binaries of this function,
- A low cost high integrity execution platform, to ensure a safe execution through the detection of any divergent behaviour.

LCHIP is also a R&D project started in Q4 2016. The tutorial is the occasion to experiment the first starter kit released by the project, including the formal IDE and the PIC32-based execution platform.



Objectives. During this tutorial, you will:

- Discover the LCHIP development environment: the IDE and its modelling language, the PIC32-based execution platform,
- Practice: model software functions, prove them, generate code, upload binaries on target hardware and execute them.

Programme:

- Introduction to LCHIP: key concepts,
- Some bits of B: introduction to the modelling language
- Programming examples: practice with several small examples

Requirements. Participants are expected to be somehow skilled in software development. Participants also have to bring their own laptop (Windows, Linux, Mac).

Tutorial 3. Model-Checking a Railway Interlocking System

Presented by Nicolas Breton from Systerel.

Nowadays, Model-Checking has become an efficient technique for demonstrating the safety of critical systems. In the railway industry it is being applied on a number of systems such as ERTMS, RBCs, and CBTCs. However, its

best successes have clearly been achieved on interlocking systems.

These systems are handling the points and signals of a railway track layout to ensure safe train operations. They hardly contain any numeric computations, but rather rely on some control-oriented formalism (plain boolean equations, automata, ...) to compute the track apparatuses commands. As they are usually obtained through the instantiation of a generic application on a given track layout, these systems can easily grow to important sizes, giving rise to huge state spaces. These simple but huge combinatorics makes them especially well fitted for model-checking.

Systerel Smart Solver (S3, www.systerel-smart-solver.com) is an industrial toolset simplifying the construction of state-of-the-art formal safety verification solutions. It is structured around a powerful Model-Checking analysis engine and its language (HLL).

An S3-based formal verification solution typically contains a custom tool to automatically model and translate to HLL the vital code of an interlocking system together with the data representing the specific track layout to which it applies. This model is then combined to a Generic Safety Specification formalized using HLL. This specification captures the high-level safety properties that an interlocking shall respect and the modeling of the operational environment of the system. It is expressed in a generic way, which, combined with the model of the track layout data, becomes specialized for this specific system. The combined model is then handed over to the S3 toolset to decide if the properties always hold or if they can be falsified by some scenarios.

The power delivered by S3 enables to address the safety in a straight-forward manner. High level safety properties of very large interlocking systems, handling hundreds of concurrent routes, are proved in a monolithic way (as opposed to compositional verification) directly on the code of the system. Moreover, by implementing a number of protection mechanisms, the S3 toolset is certifiable as a T2 EN50128:2011 verification tool.

Today, several providers and operators of both metro and main-line are routinely using S3-based solutions to verify the safety of their systems.

This tutorial proposes a comprehensive walk-through in the construction and use of an S3-based formal interlocking safety verification solution. It will address the following topics:

- basics of SAT-based Model-Checking
- constructing a formal safety verification solution
- formalizing the safety properties
- modeling the environment
- analyzing an interlocking system
- certifying the solution (EN50128:2011)

A basic knowledge of interlocking systems is required.

Tutorial 4. REVEAL: requirements elicitation, documentation and management approach

Presented by Andrew Hawthorn from Altran.

Research has shown that 48% of project failure sources are requirements problems, 41% of system errors are introduced during the requirements phase and it costs 50 times more to fix a requirements error during acceptance testing than during the requirements phase. Decades ago, academia developed methods to minimise these issues but they are still not being applied regularly in industry. In this session, we will introduce you to the core concepts of REVEAL®, which is a collection of practical, industry demonstrated techniques for producing precise, unambiguous requirements specifications that faithfully represent the needs of the key stakeholders.

REVEAL® is Altran's approach to requirements engineering, developed over two decades of practical application. It provides a rigorous Requirements Engineering framework (method, training, checklists, tool implementation ...) to ensure a complete, correct and usable set of requirements is produced and maintained.

REVEAL® addresses some of the key challenges associated with requirements engineering:

- Expression - Ensuring the correct requirements are elicited and documented to ensure we design and build the right system;
- Conflict – Recognizing that different stakeholders have different requirements and ensuring we understand which ones the system should satisfy;
- Change – Recognizing that requirements will change and ensuring that we are prepared for this.

Based on sound scientific principles and supported by best practice in model driven requirements engineering, REVEAL® is built on a combination of "soft" and "hard" skills. REVEAL is not a tool, but a methodology that has been used many times in conjunction with both IBM DOORS® and several other tools.

It has a proven delivery record on a number of significant rail programme applications, including the West Coast Main Line Upgrade, London Underground Rolling Stock, Thameslink Programme and Systems Integration and most recently the COMPASS Degraded Mode Working System.

The tutorial introduces a systematic process for applying REVEAL and covers all the key unique concepts. An interactive approach is taken, with attendee discussion encouraged, and an interactive REVEAL® “game” is included to embed ideas covered in the learning. Anyone who attends the tutorial and develops or works with requirements will be able to apply these high level concepts as soon as they return to their office.

Contact: Joan Atkinson
CSR Events Co-ordinator
Email: joan.atkinson@ncl.ac.uk
School of Computing
Newcastle University
NE4 5TG
United Kingdom
Telephone: +44 191 221 2222