Sail Through Your C Code
Either Statically or
Dynamically with MetAcsl

Virgile Robles, Nikolai Kosmatov,
Virgile Prevosto, Louis Rilling,
Pascale Le Gall

Problem
Function contracts are not suited to express every property:
• Some properties are hard to express with contracts alone
• Some properties span across a large number of functions
Lack of a high-level specification mechanism amenable to automatic verification and testing in FRAMA-C [1].

Example
Confidentiality-sensitive page management:
• each memory page has a confidentiality level
• each user has a confidentiality level
• a process can only read/write a page when allowed by the relative confidentiality levels (see Figure 1, 2)
• these constraints are pervasive in the program

Solution
A new specification mechanism: the Meta-Property
• a set of target functions
• a context (strong invariant, writing constraint, …)
• a first order property on the memory
A verification mechanism:
• translate meta-properties back to ACSL with the MetAcsl [2] plugin for FRAMA-C (see Figure 3, 4).

Contributions
• A specification mechanism, meta-properties, to express high-level properties in Frama-C, and several useful extensions.
• A specification transformation technique, enabling the use of existing assessment tools on meta-properties:
  • Static deductive verification with the WP plugin
  • Dynamic assertion checking with the E-ACSL plugin
• A FRAMA-C plugin, MetAcsl, implementing this technique automatically, making it easy to re-verify properties after a code or specification update.

References