A DSL for the safe deployment of Runtime Monitors in Cyber-Physical Systems

CISTER – Research Centre in
Real-Time & Embedded Computing Systems

Giann Spilere Nandi, David Pereira, José Proença, Eduardo Tovar
{giann, drp, pro, emt}@isep.ipp.pt

1. Background

- **Guaranteeing** that **Cyber-Physical Systems (CPS)** do **not fail** is a non-trivial problem, especially with the adoption of **complex software** solutions;
- **Formal Methods** (rigorous mathematical and logical techniques) can aid in the task of obtaining proofs about the **correctness** of CPS
  - **Runtime Verification (RV)** uses monitors and formal specifications to, during runtime, validate a set of properties.

2. The Problem

- The **deployment of RV** solutions causes an inevitable **overhead** in the system;
- If **not correctly deployed**, RV solutions can **compromise** the **security** and **safety** requirements of a system;
- **Lack of formalism** in the current state-of-the-art when it comes to the **integration** of monitoring solutions in CPS;
- Performing formal verifications is a tedious and **time-consuming** task.

3. Proposed Solution

- Apply **formal methods** to **guarantee** that deployed software monitoring solutions comply with the target system's **safety requirements**;
- Abstract formalism from the system's developer by creating a **Domain-Specific Language (DSL)** and a set of tools to support it;
- **Centralize** and **automate** formal verification procedures for the correct deployment of monitors in a given target system.

4. Proposed DSL and Associated Toolset

Our proposed DSL, and its associated toolset, must allow the developer to:

- Express **functional** and **non-functional** properties to be verified during runtime;
- **Verify**, statically, the **schedulability** of a system
  - We plan to leverage the literature on the topic of **mode-change schedulability analysis** to enforce better use of system resources;
  - We plan on supporting a **broad set** of **scheduling algorithms** and **hardware architectures**;
- **Automate** the **code generation** of the monitors and monitoring architectures following a **Correct-by-Construction** approach (i.e., complying with a set of formally defined requirements)
  - Leverage state-of-the-art formal tools like **model checkers**, **SMT solvers**, and **proof-assistants.**

Bellow, we illustrate our initial efforts to design the DSL syntax using an **automotive** electric/electronic architecture as a **use case** example:

DSL Syntax – Initial Design

```
node DC_1(scheduler = rm, cores = 2, alloc_policy = global){
  mode m_1 {
    task tsk_1{T = 10 ms, D = 10 ms, WCET = 5 ms};
    task tsk_2{T = 5 ms, D = 3 ms, WCET = 2 ms};
    monitor mon_1{T = 5 ms, D = 3 ms, WCET = 2 ms};
  }
  mode m_2 {
    task tsk_1{T = 30 ms, D = 20 ms, WCET = 15 ms};
    task tsk_2{T = 5 ms, D = 3 ms, WCET = 2 ms};
    monitor mon_1{T = 3 ms, D = 2 ms, WCET = 1 ms};
  }
}
node ECU_1 (scheduler = rm, cores = 1){
  mode m_1 {
    task tsk_3{...}; task tsk_4{...}; task tsk_5{...};
  }
  mode m_2_ {
    task tsk_5{...}; task tsk_6{...}; monitor mon_2 {...};
  }
}
```

5. Benefits

- **Enhanced** security and safety of monitored systems;
- **Reduced** development and testing time
  - Potential overall project cost reduction;
- **Centralized** tool for the deployment of monitoring solutions.

Co-financed by UIDB/04234/2020 (CISTER) and project NORTE-01-0145-FEDER-028550 (REASSURE) and project ECSEL/0016/2019 JU grant nr. 876852 (VALU3S)
FCT Fundação para a Ciência e a Tecnologia

CISTER Research Centre
ISEP, Polytechnic Institute of Porto
Rua Dr. Antº Bernardino de Almeida, 431
4249-015 Porto, Portugal

+351 228 340 502
www.cister-labs.pt
info@cister-labs.pt
facebook.com/cisterlabspt

isep Instituto Superior de Engenharia do Porto
P.PORTO