Chapter 1.3
EMBEDDED SOFTWARE AND BEYOND

Jasmin Jahić, University of Cambridge

E C S

Strategic Research and Innovation Agenda 2025

Aeneas

EPoSS. European Association on Smart Systems Integration

Inside Industry Association

Scope

- Common challenges in Embedded and cyber-physical systems - **embedded software and its engineering**
  - Challenges that arise as new applications emerge
  - Continuous integration, delivery and deployment of products and processes
  - Engineering and management during system´s entire lifecycle, including sustainability requirements

- Interoperability
- Complexity of requirements and code (safety, security, performance)
- Quality (dependability, sustainability, performance, trustworthiness)
- Lifecycle (maintainability, extendibility)
- Efficiency, effectiveness, and sustainability of software development
- Adaptability to, and the dynamic environment of ECPS
- Maintenance, integration, rejuvenation of legacy software solutions

Key trends

- Quantum Computing
- Computing accelerators
- AI
- New programming languages (Rust)
- Virtualisation and virtual prototypes

Major challenges

- Major Challenge 1: **Efficient engineering of embedded software**
- Major Challenge 2: **Continuous integration and deployment**
- Major Challenge 3: **Lifecycle management**
- Major Challenge 4: **Embedding data analytics and artificial intelligence**
- Major Challenge 5: **Support for sustainability by embedded software**
- Major Challenge 6: **Software reliability and trust**
- Major Challenge 7: **Hardware virtualization for efficient SW engineering**

R&I focus areas

New

- Digital twinning
- Constraint environments
- Continuous integration of embedded software
- Managing complexity over time
- Managing configurations over time
- Evolvability of embedded software
- Federated and distributed learning
- Embedded Intelligence
- Data streaming in constraint environments
- Embedding AI accelerators
- Resource-aware software engineering.
- Development of energy-aware and sustainable frameworks and libraries for embedded software key application areas (e.g. IoT, Smart Industry, wearables, etc.)
- Management of computation power on embedded hardware
- Composable efficient abstractions that drive sustainable solutions while optimising performance
- New concepts for programming languages to ensure quality properties by default
- Testing of systems against unexpected uses
- Standard development methods and frameworks for the development of hardware abstractions, integrated with existing tools
- Verification and validation frameworks, supported by automation, which allow for the validation of applications within virtualisation, as well as the validation of specific target systems, to confirm performance and timeliness
- Run-time environments for safety-critical applications