ARTEMIS Call 2010 Project& ARTEMIS Call 2011 Project

SafeCer

Safety Certification of Software-Intensive Systems with Reusable Components

Tooling & Interoperability

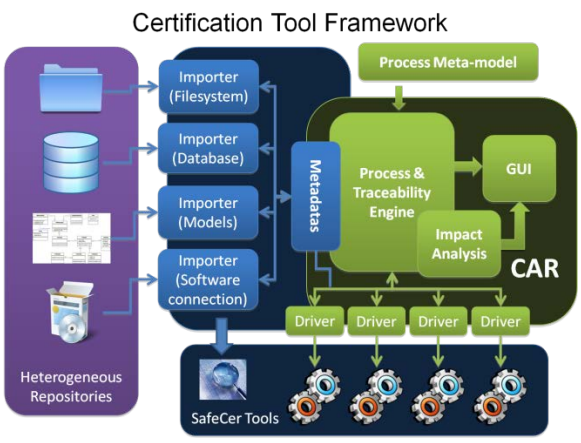
TARGET TOOL SUPPORT

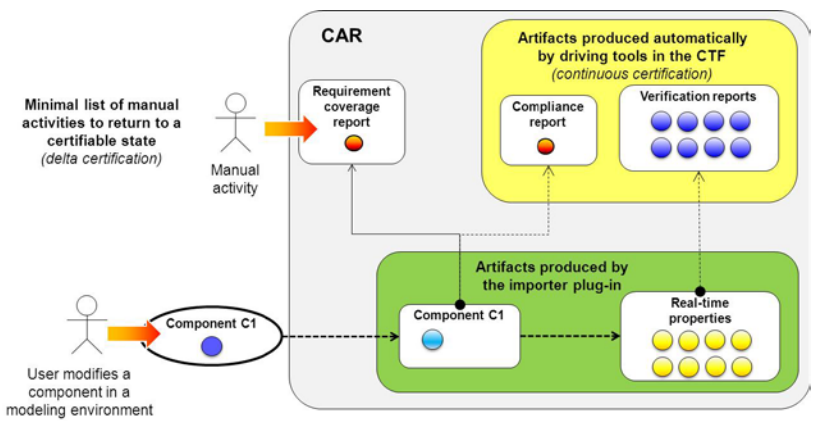
SafeCer is developing a tool framework that supports:

- An integrated certification and development process in accordance with the different safety standards.
- Compositional certification and verification by means of
  - Modeling of system and software components.
  - Specification and verification of component contracts and other safety-related properties.
  - Collection of all certification and development artifacts to generate the safety case.

INTEROPERABILITY FOR CONTINUOUS CERTIFICATION

SafeCer focuses on developing the domain-independent part of tools, integrating them, and providing the mechanism for seamless interoperability among the tools. This will permit to check for artifacts metadata consistency, end-to-end traceability and compliance to process along with their user interface. Furthermore additional tools are planned, such as facilities to automate the execution of certification-oriented activities (such as test execution and collection of verification reports) so as to achieve continuous certification.





Certification Tool Framework

SafeCer provides a generic domain-independent Certification Tool Framework (CTF) which uses different tools in an integrated environment and can be tailored (instantiated) for the domain-specific requirements defined in the applicable safety standards.

Fundamental bricks of the CTF are:

- The interoperability platform to launch different tools for acquiring and processing the certification and development artifacts. Based on the DISC platform, it provides an xml format to exchange meta-data about the artifacts and check their consistency. **SafeCer has started activities to use** as an option **the evolving ARTEMIS IOS approach with OSLC** (as further developed at the moment particularly in **CRYSTAL).**
- The Certification Artifact Repository (CAR) to store all certification and development artifacts, trace their dependencies, and control the process. Based on this information, it can identify a minimal set of activities to return to a certifiable state.
- The Workflow Engine for Analysis, Certification and Test (WEFACT) to control that the flow of activities respects the certification and development process instantiated for specific standards and to generate the related safety cases. **WEFACT** will implement the **IOS/OSLC** option.
- Modeling tools (such as Papyrus-MDT, CHESS, VERDE) to support the specification of system and software components, their contracts and other certification-related properties.
- Validation and verification tools (such as NuSMV3) to verify that component contracts are correctly derived and implemented.
- Other heterogeneous tools for requirements, verification, and certification (such as Magillem Suite, TTEVerify, and SADD) used in specific demonstrators.

More tools can be added by just developing the driver and the importer to handle artifacts' meta-data. Tool chains are enhanced by the CTF with end-to-end traceability (although they must exist independently).