SIMULUS Projects

The projects composing SIMULUS can be split in the following categories:

- **Support**: these projects provide the ECSS SMP model based approach and development framework.
- **Models**: these projects provide the SMP Models that are typically used to build a simulator.
- **Runtime**: these projects provide the software used to execute SMP based simulations.

Support Projects

UMF

The UMF project (Unified Modelling Framework) provides the different processes needed in the Model-Driven approach for the creation of ECSS SMP models. The main features provided are:

1. UML design for ECSS SMP modelling the MagicDraw Software, including:
    - UML profile for the MagicDraw tool for ECSS SMP modelling.
    - Import of requirements into a MagicDraw design file.
    - Export of UML XMI from a MagicDraw design file.
    - Export of diagrams from a MagicDraw design file.
2. Code and Documentation generation from design, including:
    - ECSS SMP artefacts generation from UML XMI.
    - C++ Code generation from ECSS SMP artefacts.
    - Documentation generation from ECSS SMP artefacts.

NOTE: Use of UMF for Code and Documentation generation does not require strictly Magicdraw but the Eclipse `uml` files. However since only the MagicDraw `mdxml` files are commited to the repositories to generate code and documentation, one has to have MagicDraw available at least once to allow the generation of the `uml` files.

Build System

The Build System project provides a generic build system that is used in the rest of the SIMULUS projects but also supports the development of SIMULUS based simulators. Its main features are:

- Integration with Maven for build system orchestration, binary packages creation and dependency management.
- Simple support for Java based projects.
- Extended support for UMF based C++ projects, including:
  - Simple management of ECSS SMP libraries creation.
  - Configuration of dependencies necessary for SIMULUS.
  - Definition and execution of unit, integration and system tests.

Documentation

The Documentation project includes the configuration files used for the creation of the SIMULUS reference documentation site available at https://sim.space-codev.org/docs.

SIMULUS Product Bundle

The SIMULUS Product project includes the top-level documentation pages for the SIMULUS reference documentation (e.g. CiG) as well as other useful SIMULUS level tools.

Containers

The Containers project contains Docker container files for SIMULUS runtime and build environments.

Pipelines

The Pipelines project contains Gitlab CI Pipeline files for building the full set of SIMULUS projects.

Models

ECSS SMP

The ECSS SMP project contains the base models supporting ECSS SMP development. This includes:

- ECSS SMP header files
- SIMULUS extensions to ECSS SMP headers
- A Component Development Kit with base implementations of ECSS SMP concepts
- Test Harness for both C++ and Java projects
- Generic utilities used in the rest of the SIMULUS Models projects
- A verification of the support for ECSS-SMP code generation with UMF (`VerSim`)

Generic Models

The Generic Models project contains a large set of models based on ECSS SMP covering different areas of a spacecraft simulator, such as:

- Electrical Subsystem
- Telemetry and Telecommand encoding and decoding
- Position and environment modelling
- Thermal Network simulation
- Telecommunication Subsystem
- Communication with Ground Systems

Emulator

The Emulator project provides a powerful Emulator for the ERC32, LEON2 and LEON3 processors as well as some accompanying devices, including co-processor and floating-point instructions.

Reference Architecture

The Reference Architecture project provides a set of interfaces designed with the objective to define a common architecture for operational simulators that facilitates model reuse.

Reference Architecture Simulator

The Reference Architecture Simulator project provides a sample implementation of the Reference Architecture using models from the Generic Models project.

Runtime

SIMSAT Kernel

The SIMSAT Kernel project provides an ECSS-SMP compliant simulation kernel that can be used to run simulators and interact with them.

SIMSAT MMI

The SIMSAT MMI project is a graphical front-end for controlling and monitoring a simulation running in SIMSAT Kernel.