

**OHB**

**RATIO-SIM Position**



# OHB Statements

## No need for Rationalization?

SMP has proven that a standard does not preclude the development of specific solutions tailored to particular problem domains (e.g. SVF, AIVS, TOMS, ...). It rather enables sharing/reuse between those solutions. Therefore well defined rationalization with clear scope has very obvious advantages.

## Rationalization too costly?

If an incremental upgrade / replacement path for individual parts of the development process is available, migration costs can be minimized; basing new solutions on future rationalization and standardization work beyond SMP would give confidence that those new solutions are 'future-proof'.

## Scoping?

Opportunities for future rationalization exist along the entire development process: e.g. standardized test/validation processes and documentation generation.

## Problem already solved? No need for a harmonised / consolidated Simulation Infrastructure?

SMP and its evolution do solve a (considerable) part of the problem. But there is no reason to stop here – see e.g. EGS-CC and many of the talks at SESP 2017...

## Spin-in not possible?

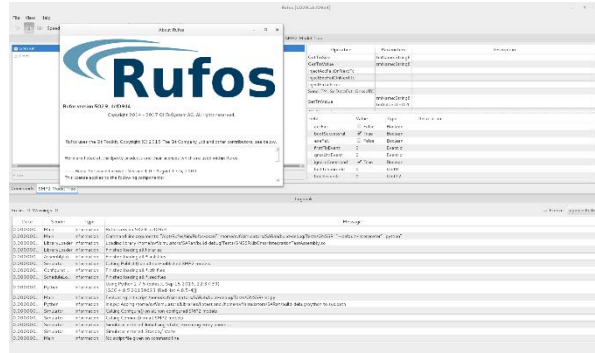
Not true; e.g. new state-of-the-art software development tools and processes are being introduced all the time (and e.g. the new ECSS SMP standard moves to C++11). Standardization work must keep up with new (software) technologies.

# OH B Statements

Current status:

## OH B Software Base Simulator:

SMP2 runtime environment Rufos, Platform and Common Models are shared between projects:



### Platform Models

Generic functionalities that recur within different simulator facilities, e.g.

- Line Models:
  - MilBus,
  - SpaceWire
  - CAN etc..
- Debug Interface and more...

### Common Models

Models of standardized S/C hardware used in various missions

SMU / OBC

μRTU

GNSSR

Future: Are the European Industry (Primes and SMEs) willing to work towards a harmonised Modelling and Simulation Infrastructure? **Yes** - preferably modular approaches: single/well defined improvement projects/software tools: define small projects that could improve specific parts of tools which are used during development and review (e.g. doc generation tool)