

# Capital Software Designer

## Streamlining model-based software engineering

### Benefits

- Create enriched architecture models to drive the design process
- Connect to your implementation tool of choice to develop code
- Integrate, verify and validate your code to deliver onboard software quality

### Summary

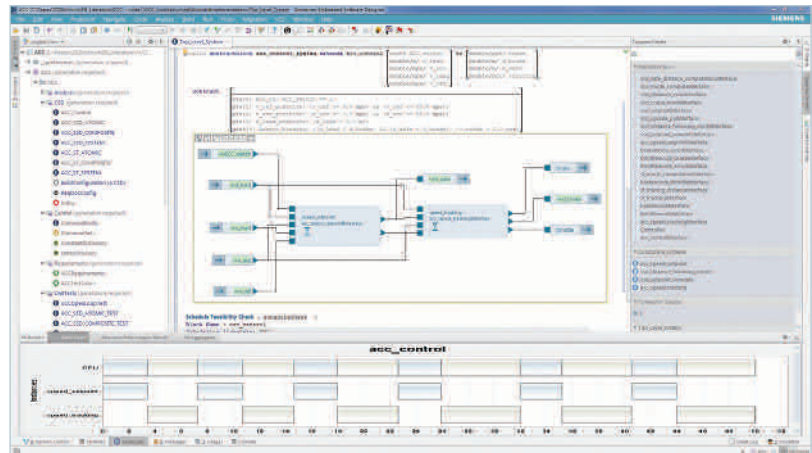
Capital Software Designer software connects the elements of your embedded software design processes and helps you avoid rework costs by detecting errors early in the process. An architecture-centric approach allows you to define, enrich, analyze and simulate onboard software designs.

The solution is open so you can implement onboard software designs in external development environments with the template export functions. Its external development support covers

the C language for hand coding and the Simulink® environment interface for model-based development.

The Capital Software Designer contract-based architecture enables you to drive testing, verification and closed-loop simulation, even when following mixed external implementation paradigms. Capital Software Designer is interoperable with other validation tools and platforms, thus providing digital continuity in scattered process and tool landscapes.

Capital Software Designer is tightly integrated with other Simcenter solutions, as well as Polarion ALM™ software. This provides a connected environment for embedded system and software architecting, engineering, design, validation as well as multi-physics simulation, application testing and lifecycle management.



# Capital Software Designer

Name	Kind	Type	Unit	Constraints	Description
temperature	quantity<name>	double	degC	range -100 degC .. 250 degC	Generic temperature
ambientTemp	quantity< temperature>			range -50 degC .. 55 degC	Ambient temperature
batteryTemp	quantity< temperature>			range -50 degC .. 95 degC	Battery temperature
temperatureError	quantity< temperature>			range -20 degC .. 48 degC	Difference between battery temperature and desired temper
cabinTemp	quantity< temperature>			range -50 degC .. 55 degC	Cabin temperature
referenceCabinTemp	quantity< temperature>			range 18 degC .. 32 degC	Desired cabin temperature
referenceBatteryTemp	quantity< temperature>			range 20 degC .. 30 degC	Desired battery temperature
vehicleSpeed	quantity<name>	double	km/h	range -50 km/h .. 300 km/h	Vehicle linear speed
airSpeed	quantity<name>	double	km/h	range -20 km/h .. 120 km/h	Net air speed over the vehicle [air speed opposite to drive + vehicle speed]
airMassFlowRate	quantity<name>	double	kg/s	range 0 kg/s .. 0.85 kg/s	Air mass flow rate in/out of the cabin
freshAirMassFlowRate	quantity< airMassFlowRate>			<no constraints>	Fresh air mass flow rate into the cabin
recircAirMassFlowRate	quantity< airMassFlowRate>			<no constraints>	Recirc air mass flow rate within the cabin

- Identify precise type system and physical units
- Import and merge the architecture created in Systems Modeling Language (SysML), Autosar and Architecture Analysis and Design Language (AADL)
- Use intuitive graphical modeling based on unlimited nesting of data flows
- Re-use test case definitions to help clarify requirements and functional specifications

## Streamlining model-based development

Create a holistic software architecture that acts as single source of truth across the software development lifecycle.

## Capture and enrich software architecture

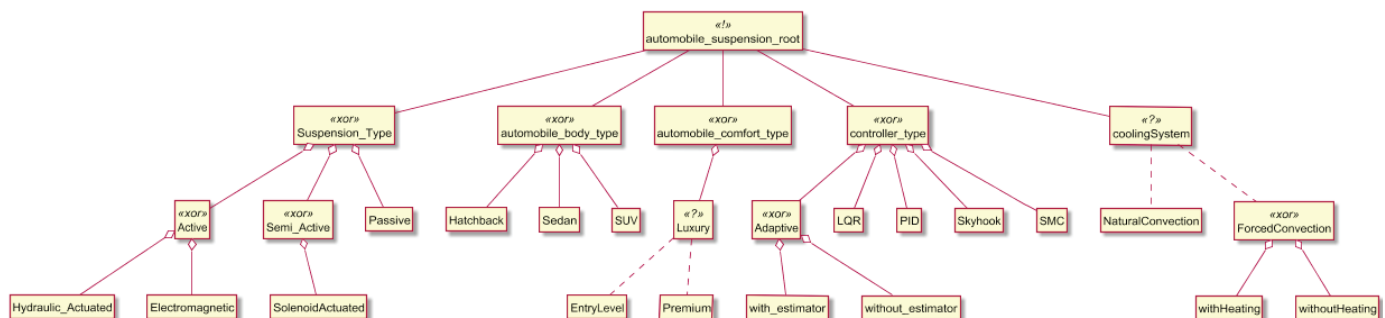
- Use data-flow modeling to support abstraction and re-use
- Enrich software architectures with contract and timing needs

## Analyze software architectures

- Analyze architecture for consistency, completeness and early detection of specification errors enabled by formal methods
- Schedule feasibility analysis to front-load software-hardware co-development

## Manage design data

- Use detailed specification of elements in the data dictionary to specify data types, physical units and range constraints for signals and parameters
- Capture the product line variability in the parametric data by using constant groups





simulation with plant models developed in Simcenter Amesim™ software and Simulink

- Export Functional Mock-up Units (FMUs) with choice of compiler (GCC/VCC) and operating platform (32/64 bits) for connecting to Simcenter System Synthesis and any Functional Mock-up Interface (FMI) compliant system simulation tools

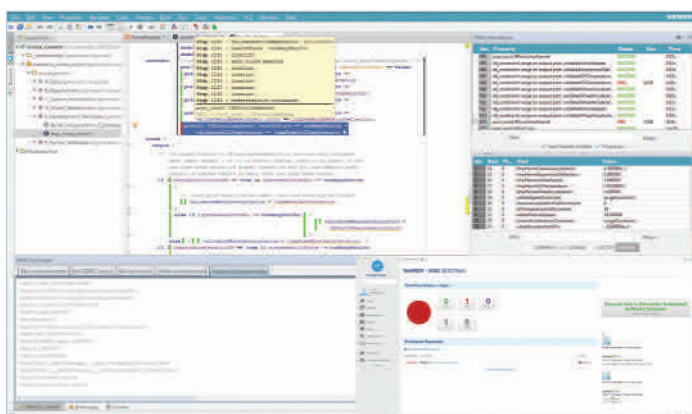
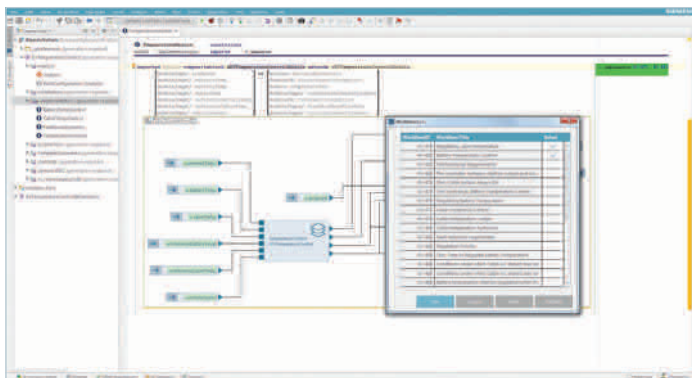
### Verify using CBMC

- Verify key runtime properties; division-by-zero, array-out-of-bounds, memory leak, arithmetic overflow and pointer dereferencing
- Verify safety properties expressed in the form of contracts, assertions and data dictionary constraints
- Analyze dead and unreachable code

- Use automatic counter example production at the level of model and line of code
- Increase coverage with code coverage measurement and automatic generation of test vectors

### Prepare target deployment of application code

- Analyze worst-case execution time (WCET) independently of the target platform
- Estimate program and data memory and compare the old and new result after a change



Siemens Digital Industries Software  
www.sw.siemens.com

Americas +1 314 264 8499  
Europe +44 (0) 1276 413200  
Asia-Pacific +852 2230 3333