MÄLARDALEN UNIVERSITY
SWEDEN

AMASS

Variant Management and Change Impact Analysis in Safety-oriented Process-Product Lines

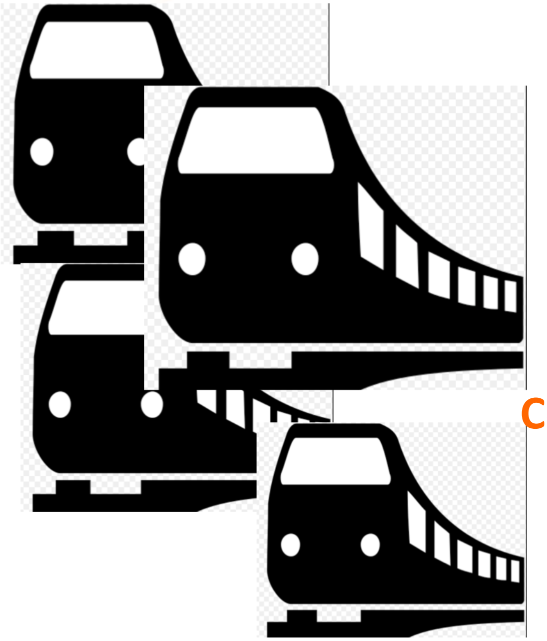
Barbara Gallina
barbara.gallina@mdh.se
Certifiable Evidences & Justification Engineering
Mälardalen University, Västerås, Sweden

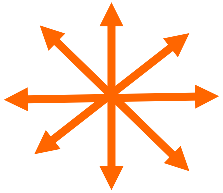
M. A. Javed, **B. Gallina**, and Anna Carlsson
Towards Variant Management and Change Impact Analysis in Safety-oriented Process-Product Lines.
Poster-paper at 34th ACM/SIGAPP Symposium On Applied Computing (**SAC**).
Limassol, Cyprus April 8-12, ACM, New York, NY, USA, 2372-2375.
DOI: https://doi.org/10.1145/3297280.3297634, 2019

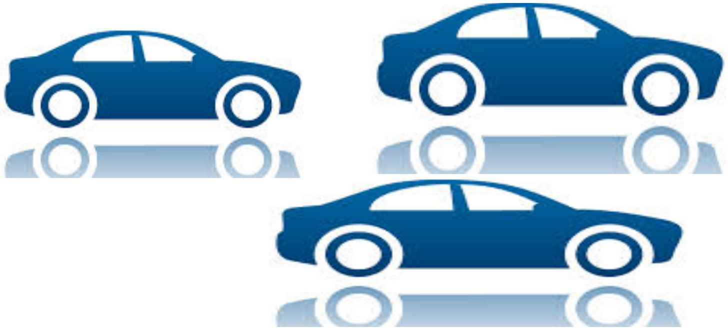
AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems)

7th Scandinavian Conference on SYSTEM & SOFTWARE SAFETY, October 23rd, 2019

Context and Motivation







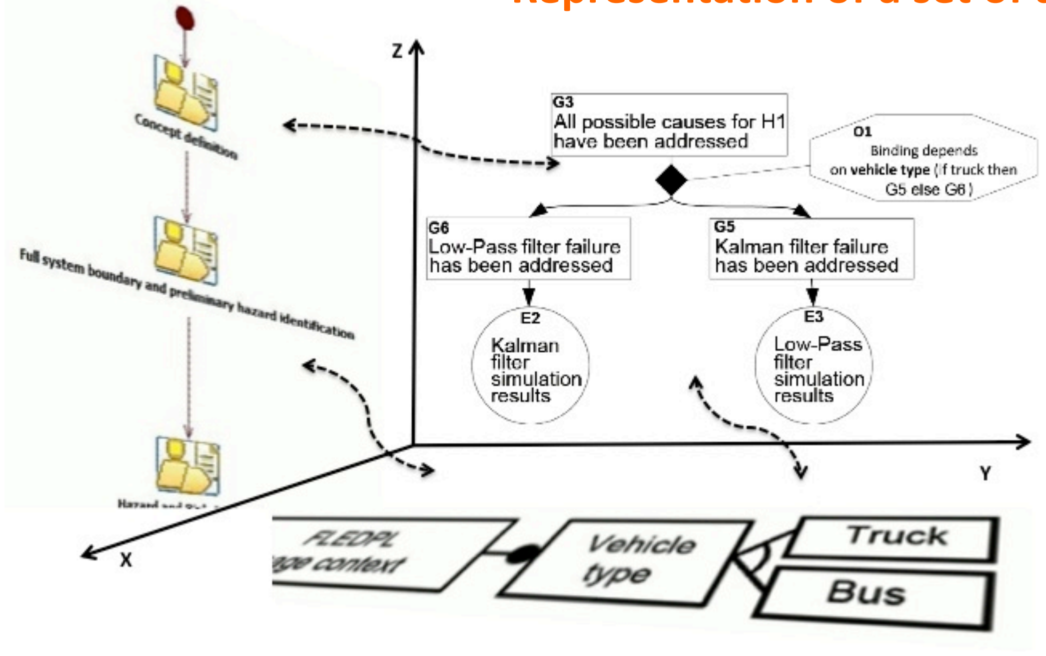
Commonalities and variabilities systematization/management/reuse

Different hazards?
Different classification?
->different product/process/assurance case

..towards a solution

Representation of a set of processes

Representation of a set of assurance cases


Z
Concept definition
Full system boundary and preliminary hazard identification
Hazard
G3
All possible causes for H1 have been addressed
O1
Binding depends on vehicle type (if truck then G5 else G6)
G6
Low-Pass filter failure has been addressed
G5
Kalman filter failure has been addressed
E2
Kalman filter simulation results
E3
Low-Pass filter simulation results
Y
X
FLEDPL
ge context
Vehicle type
Truck
Bus

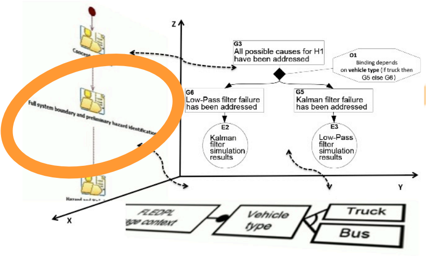
-> variation points and interdependencies

Representation of a set of Product

B. Gallina. Towards Enabling Reuse in the Context of Safety-Critical Product Lines. In 5th IEEE/ACM International Workshop on Product Line Approaches in Software Engineering, PLEASE 2015, Florence, Italy, 15–18 May 19, 2015

Outline

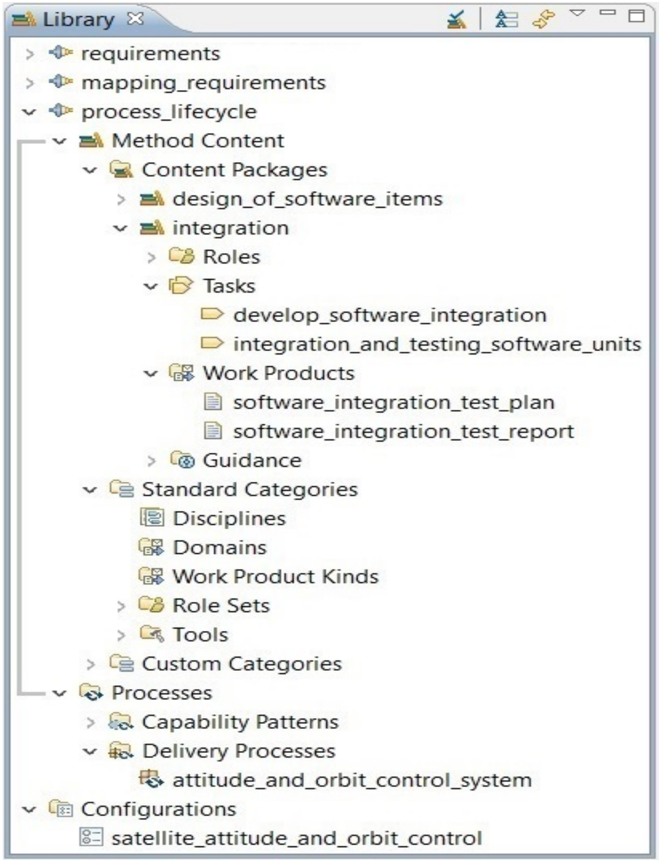
- Background
  - EPF Composer
  - CHESS Toolset
  - BVR Tool
- Managing process/product variability
- Illustrative Example
- Future Work


Concept
Full system boundary and preliminary hazard identification
Hazard and Risk
Z
Y
X
G3
All possible causes for H1 have been addressed
O1
Binding depends on vehicle type (if truck then G5 else G6)
G6
Low-Pass filter failure has been addressed
G5
Kalman filter failure has been addressed
E2
Kalman filter simulation results
E3
Low-Pass filter simulation results
FLEDPL
usage context
Vehicle type
Truck
Bus

EPF Composer

- Eclipse Process Framework (EPF) Composer[1] provides an extensible framework and tooling for authoring, configuring and publishing processes

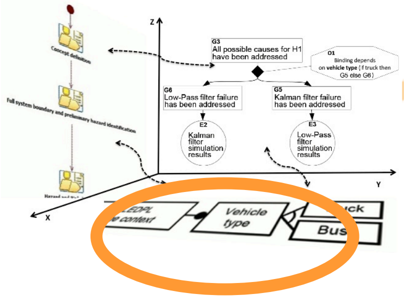
- EPF Composer is based on the Unified Method Architecture (UMA) metamodel that supports major parts of the Software & Systems Process Engineering metamodel (SPEM) 2.0[2]



- requirements
- mapping_requirements
- process_lifecycle
  - Method Content
    - Content Packages
      - design_of_software_items
      - integration
        - Roles
        - Tasks
          - develop_software_integration
          - integration_and_testing_software_units
        - Work Products
          - software_integration_test_plan
          - software_integration_test_report
        - Guidance
    - Standard Categories
      - Disciplines
      - Domains
      - Work Product Kinds
      - Role Sets
      - Tools
    - Custom Categories
  - Processes
    - Capability Patterns
    - Delivery Processes
      - attitude_and_orbit_control_system
- Configurations
  - satellite_attitude_and_orbit_control

[1] http://www.eclipse.org/epf/

[2] http://www.omg.org/spec/SPEM/2.0/


Concept definition
Full system boundary and preliminary hazard identification
Hazard and risk
Z
Y
X
G3 All possible causes for H1 have been addressed
O1 Binding depends on vehicle type [if truck then G5 else G6]
G6 Low-Pass filter failure has been addressed
G5 Kalman filter failure has been addressed
E2 Kalman filter simulation results
E3 Low-Pass filter simulation results
EDPL context
Vehicle type
ck
Bus

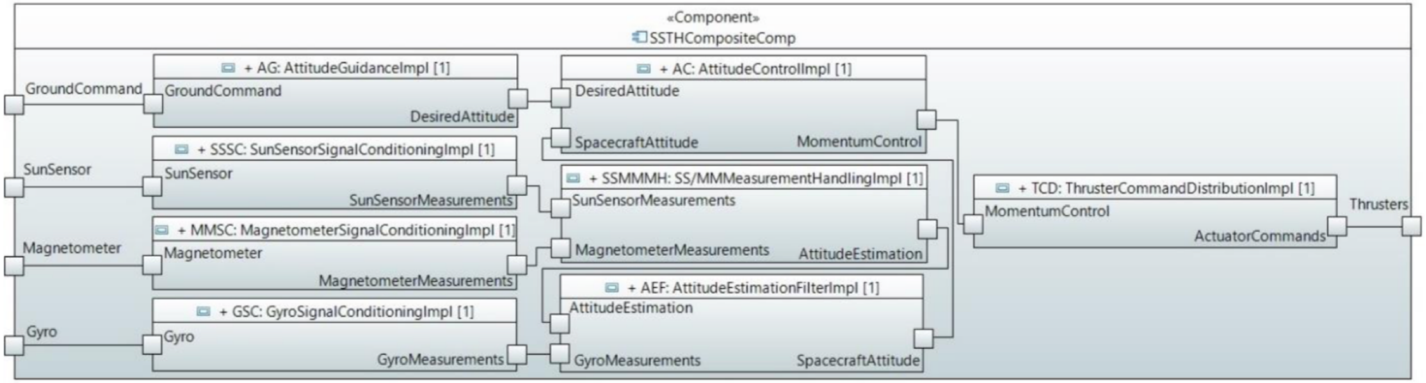
CHESS Toolset

- Open-source tooset implementing the CHESS methodology

CHESS

Composition with Guarantees for High-integrity Embedded Software Components Assembly

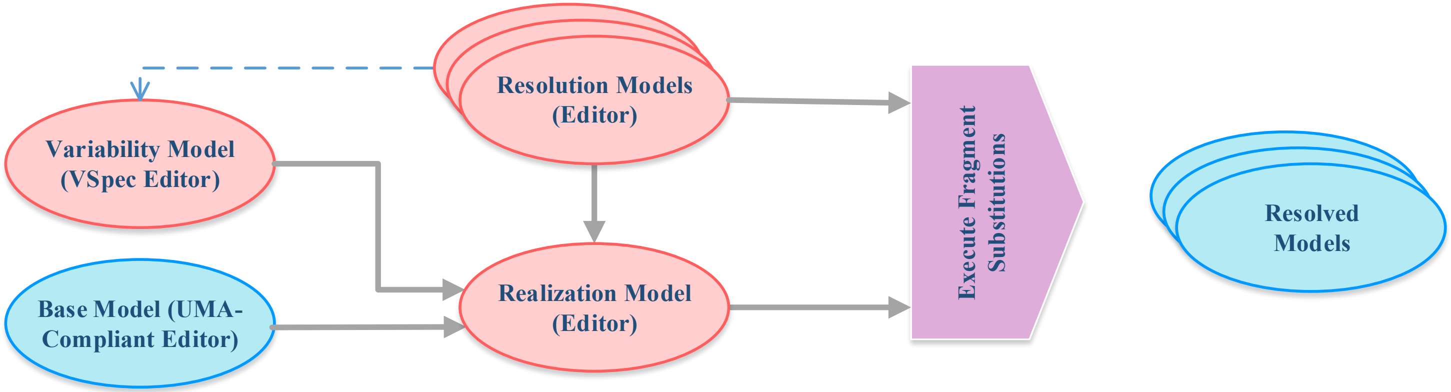
https://www.polarsys.org/projects/polarsys.chess



AMASS, Deliverable D6.3, link:
https://www.amass-ecsel.eu/sites/amass.drupal.pulsartecnalia.com/files/documents/
D6.3_Design-of-the-AMASS-tools-and-methods-for-cross-intra-domain-reuse-%28b%29_AMASS_Final.pdf

BVR Tool

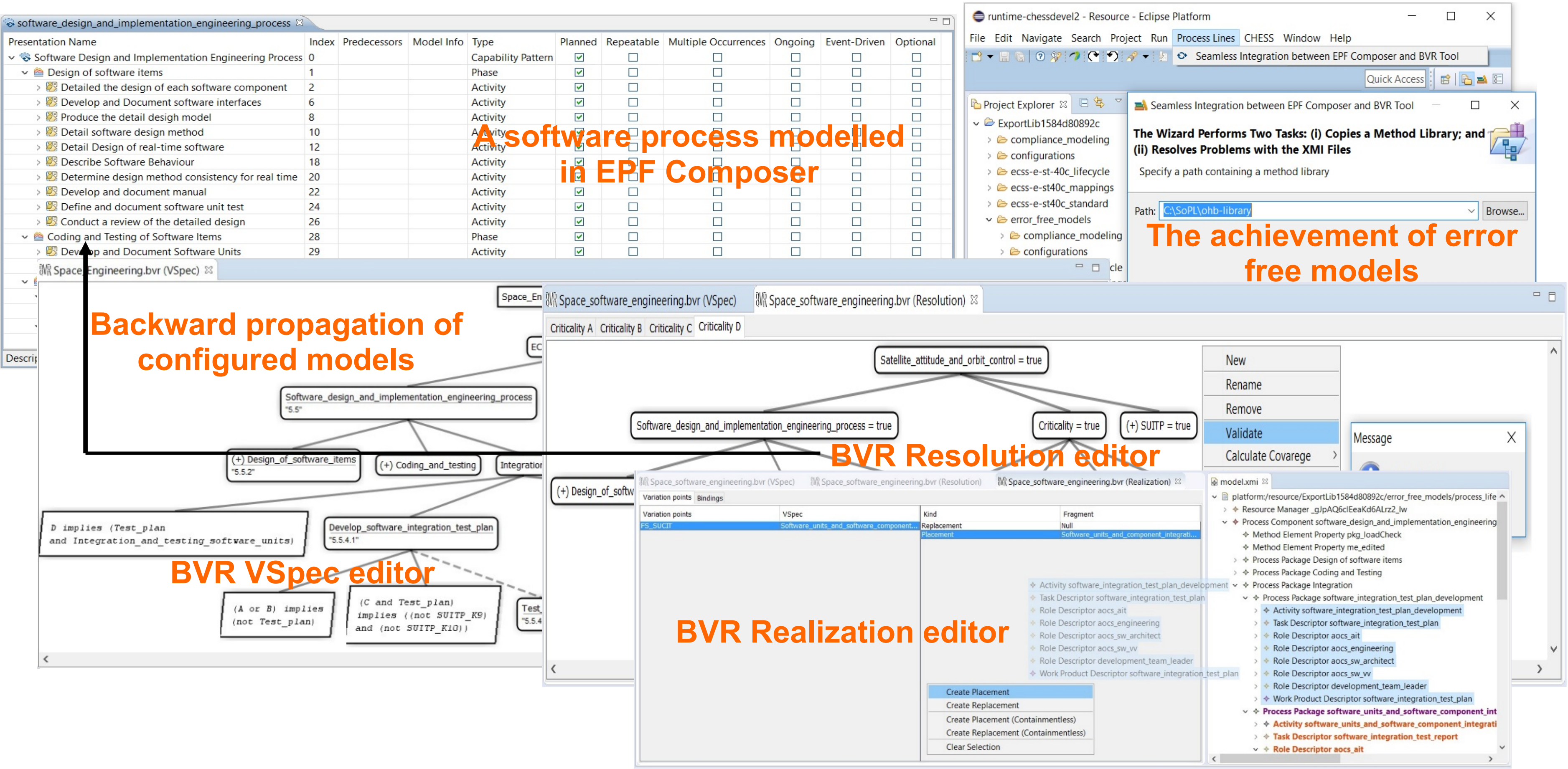
- The Base Variability Resolution (BVR) Tool is an implementation of the Common Variability Language (CVL) standard tailored for the necessities of the VARIES[3] project. It enables:
  - Feature Modeling/**abstract representation** (VSpec)
  - Feature inclusion/exclusion (Resolution model)
  - Abstract representation and concrete representation binding (Realization model)



[3] http://www.varies.eu

Variability management at process level

-Illustrative Example-



software_design_and_implementation_engineering_process

Presentation Name	Index	Predecessors	Model Info	Type	Planned	Repeatable	Multiple Occurrences	Ongoing	Event-Driven	Optional
Software Design and Implementation Engineering Process	0			Capability Pattern	☑	□	□	□	□	□
Design of software items	1			Phase	☑	□	□	□	□	□
Detailed the design of each software component	2			Activity	☑	□	□	□	□	□
Develop and Document software interfaces	6			Activity	☑	□	□	□	□	□
Produce the detail design model	8			Activity	☑	□	□	□	□	□
Detail software design method	10			Activity	☑	□	□	□	□	□
Detail Design of real-time software	12			Activity	☑	□	□	□	□	□
Describe Software Behaviour	18			Activity	☑	□	□	□	□	□
Determine design method consistency for real time	20			Activity	☑	□	□	□	□	□
Develop and document manual	22			Activity	☑	□	□	□	□	□
Define and document software unit test	24			Activity	☑	□	□	□	□	□
Conduct a review of the detailed design	26			Activity	☑	□	□	□	□	□
Coding and Testing of Software Items	28			Phase	☑	□	□	□	□	□
Develop and Document Software Units	29			Activity	☑	□	□	□	□	□

A software process modelled in EPF Composer

runtime-chessdevel2 - Resource - Eclipse Platform

Seamless Integration between EPF Composer and BVR Tool

The Wizard Performs Two Tasks: (i) Copies a Method Library; and (ii) Resolves Problems with the XMI Files

Specify a path containing a method library

Path: C:\SoPL\ohb-library Browse...

The achievement of error free models

Space_Engineering.bvr (VSpec)

Backward propagation of configured models

Software_design_and_implementation_engineering_process
"5.5"
(+) Design_of_software_items
"5.5.2"
(+) Coding_and_testing
Integration
D implies (Test_plan and Integration_and_testing_software_units)
Develop_software_integration_test_plan
"5.5.4.1"
(A or B) implies (not Test_plan)
(C and Test_plan) implies ((not SUITP_K9) and (not SUITP_K10))
Test_
"5.5.4

BVR VSpec editor

Space_software_engineering.bvr (VSpec) | Space_software_engineering.bvr (Resolution)

BVR Resolution editor

Space_software_engineering.bvr (VSpec) | Space_software_engineering.bvr (Resolution) | Space_software_engineering.bvr (Realization)

Variation points | Bindings

Variation points	VSpec	Kind	Fragment
FS_SUCIT	Software_units_and_software_component...	Replacement	Null
		Placement	Software_units_and_component_integrati...

Activity software_integration_test_plan_development
Task Descriptor software_integration_test_plan
Role Descriptor aocs_ait
Role Descriptor aocs_engineering
Role Descriptor aocs_sw_architect
Role Descriptor aocs_sw_vv
Role Descriptor development_team_leader
Work Product Descriptor software_integration_test_plan
Create Placement
Create Replacement
Create Placement (Containmentless)
Create Replacement (Containmentless)
Clear Selection

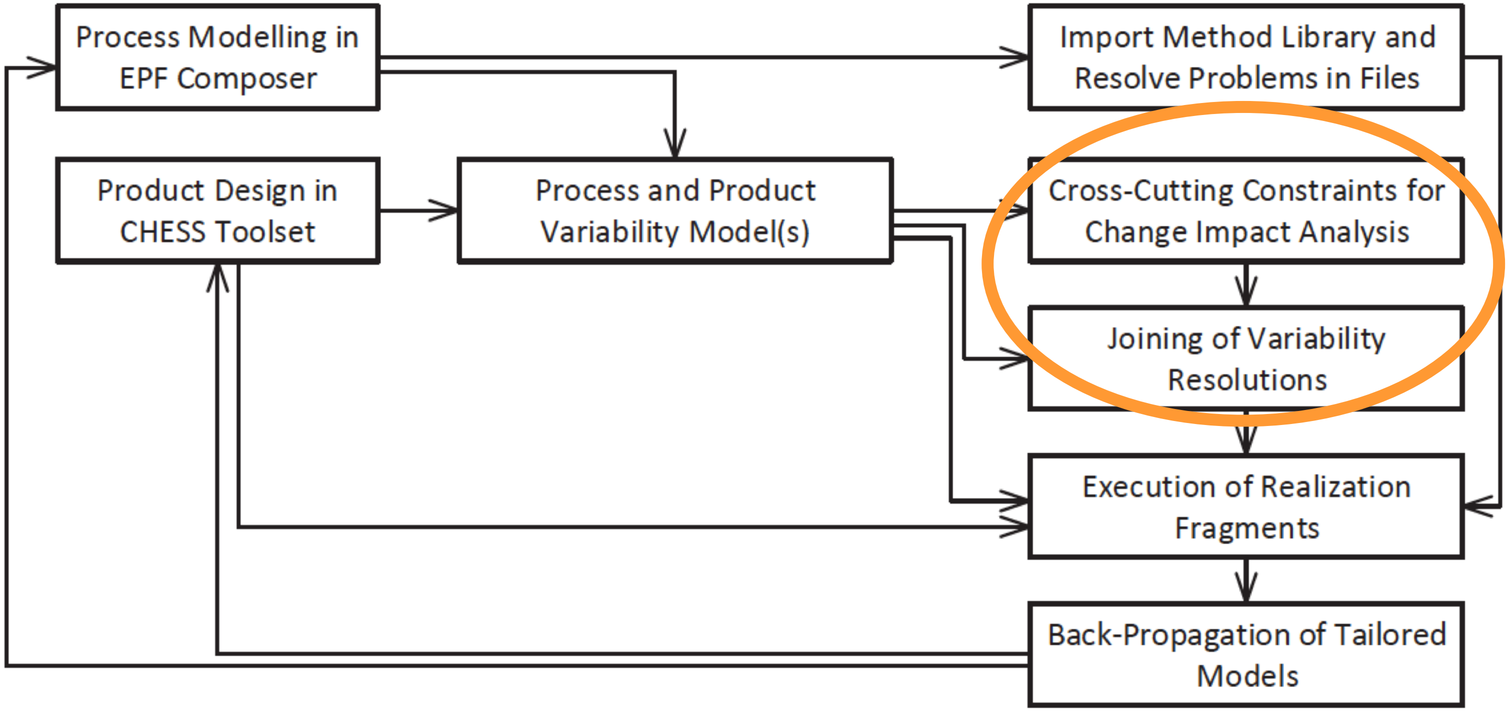
BVR Realization editor

model.xmi

- platform:/resource/ExportLib1584d80892c/error_free_models/process_life
  - Resource Manager _gJpAQ6clEeaKd6ALrz2_lw
  - Process Component software_design_and_implementation_engineering
    - Method Element Property pkg_loadCheck
    - Method Element Property me_edited
    - Process Package Design of software items
    - Process Package Coding and Testing
    - Process Package Integration
      - Process Package software_integration_test_plan_development
        - Activity software_integration_test_plan_development
        - Task Descriptor software_integration_test_plan
        - Role Descriptor aocs_ait
        - Role Descriptor aocs_engineering
        - Role Descriptor aocs_sw_architect
        - Role Descriptor aocs_sw_vv
        - Role Descriptor development_team_leader
        - Work Product Descriptor software_integration_test_plan
      - Process Package software_units_and_software_component_int
        - Activity software_units_and_software_component_integrati
        - Task Descriptor software_integration_test_report
        - Role Descriptor aocs_ait

M. A. Javed and B. Gallina. Safety-oriented Process Line Engineering via Seamless Integration between EPF Composer and BVR Tool. In 22nd International Systems and Software Product Line Conference (SPLC), Sept 10-14, Gothenburg, Sweden, ACM Digital Library, DOI: 10.1145/3236405.3236406, 2018.

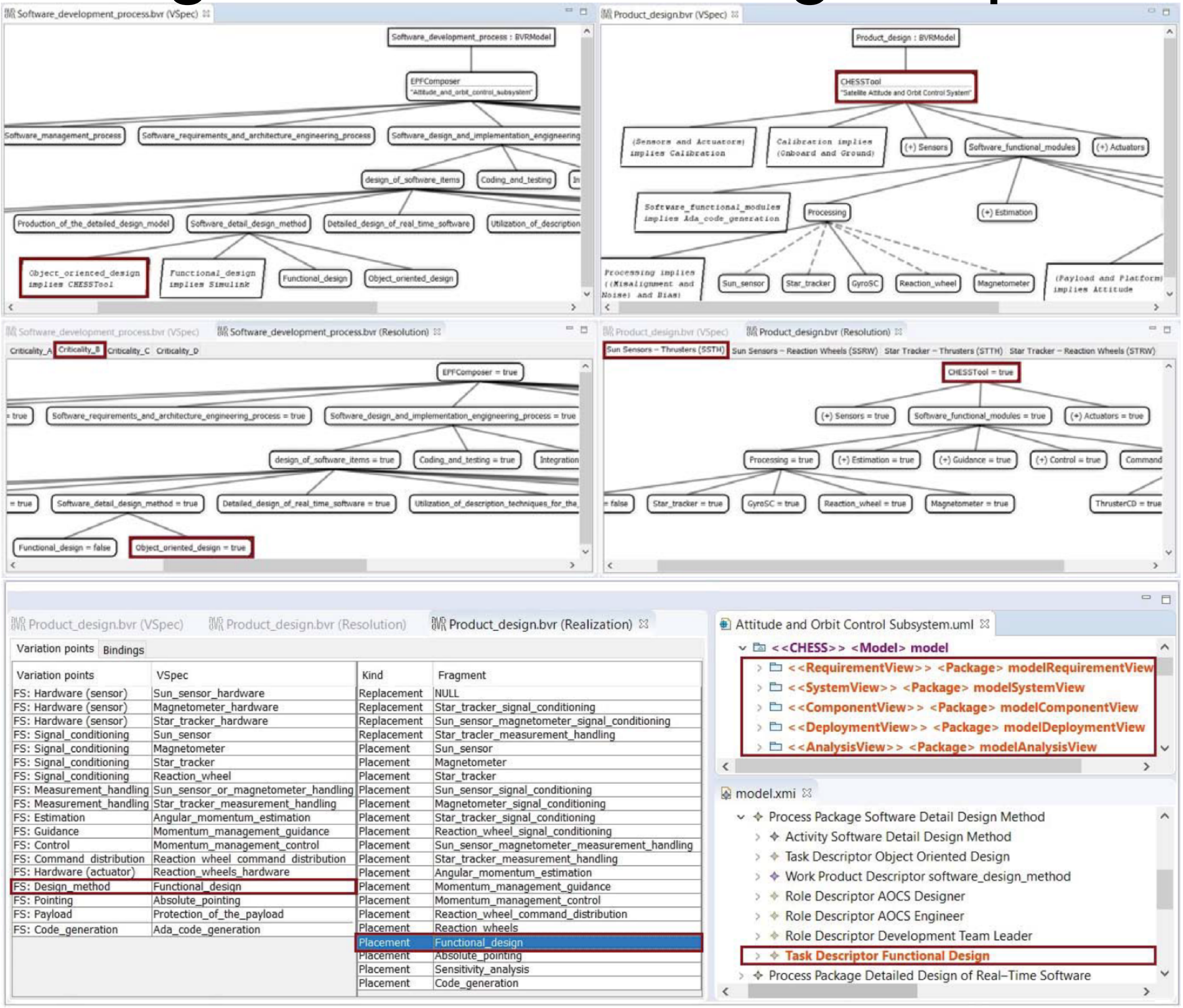
Variant Management and Change Impact Analysis


Process Modelling in EPF Composer
Import Method Library and Resolve Problems in Files
Product Design in CHESS Toolset
Process and Product Variability Model(s)
Cross-Cutting Constraints for Change Impact Analysis
Joining of Variability Resolutions
Execution of Realization Fragments
Back-Propagation of Tailored Models

MÄLARDALEN UNIVERSITY SWEDEN

AMASS

Variant Management and Change Impact Analysis


Software_development_process.bvr (VSpec)
Software_development_process : BVRModel
EPFComposer
"Attitude_and_orbit_control_subsystem"
Software_management_process
Software_requirements_and_architecture_engineering_process
Software_design_and_implementation_engigneering
design_of_software_items
Coding_and_testing
In
Production_of_the_detailed_design_model
Software_detail_design_method
Detailed_design_of_real_time_software
Utilization_of_description
Object_oriented_design implies CHESSTool
Functional_design implies Simulink
Functional_design
Object_oriented_design
Product_design.bvr (VSpec)
Product_design : BVRModel
CHESSTool
"Satellite Attitude and Orbit Control System"
(Sensors and Actuators) implies Calibration
Calibration implies (Onboard and Ground)
(+) Sensors
Software_functional_modules
(+) Actuators
Software_functional_modules implies Ada_code_generation
Processing
(+) Estimation
Processing implies ((Misalignment and Noise) and Bias)
Sun_sensor
Star_tracker
GyroSC
Reaction_wheel
Magnetometer
(Payload and Platform) implies Attitude
Software_development_process.bvr (VSpec)
Software_development_process.bvr (Resolution)
Criticality_A
Criticality_B
Criticality_C
Criticality_D
EPFComposer = true
= true
Software_requirements_and_architecture_engineering_process = true
Software_design_and_implementation_engigneering_process = true
design_of_software_items = true
Coding_and_testing = true
Integration
= true
Software_detail_design_method = true
Detailed_design_of_real_time_software = true
Utilization_of_description_techniques_for_the
Functional_design = false
Object_oriented_design = true
Product_design.bvr (VSpec)
Product_design.bvr (Resolution)
Sun Sensors – Thrusters (SSTH)
Sun Sensors – Reaction Wheels (SSRW)
Star Tracker – Thrusters (STTH)
Star Tracker – Reaction Wheels (STRW)
CHESSTool = true
(+) Sensors = true
Software_functional_modules = true
(+) Actuators = true
Processing = true
(+) Estimation = true
(+) Guidance = true
(+) Control = true
Command
= false
Star_tracker = true
GyroSC = true
Reaction_wheel = true
Magnetometer = true
ThrusterCD = true

Product_design.bvr (VSpec) | Product_design.bvr (Resolution) | Product_design.bvr (Realization)

Variation points | Bindings

Variation points	VSpec	Kind	Fragment
FS: Hardware (sensor)	Sun_sensor_hardware	Replacement	NULL
FS: Hardware (sensor)	Magnetometer_hardware	Replacement	Star_tracker_signal_conditioning
FS: Hardware (sensor)	Star_tracker_hardware	Replacement	Sun_sensor_magnetometer_signal_conditioning
FS: Signal_conditioning	Sun_sensor	Replacement	Star_tracler_measurement_handling
FS: Signal_conditioning	Magnetometer	Placement	Sun_sensor
FS: Signal_conditioning	Star_tracker	Placement	Magnetometer
FS: Signal_conditioning	Reaction_wheel	Placement	Star_tracker
FS: Measurement_handling	Sun_sensor_or_magnetometer_handling	Placement	Sun_sensor_signal_conditioning
FS: Measurement_handling	Star_tracker_measurement_handling	Placement	Magnetometer_signal_conditioning
FS: Estimation	Angular_momentum_estimation	Placement	Star_tracker_signal_conditioning
FS: Guidance	Momentum_management_guidance	Placement	Reaction_wheel_signal_conditioning
FS: Control	Momentum_management_control	Placement	Sun_sensor_magnetometer_measurement_handling
FS: Command distribution	Reaction wheel command distribution	Placement	Star_tracker_measurement_handling
FS: Hardware (actuator)	Reaction_wheels_hardware	Placement	Angular_momentum_estimation
FS: Design_method	Functional_design	Placement	Momentum_management_guidance
FS: Pointing	Absolute_pointing	Placement	Momentum_management_control
FS: Payload	Protection_of_the_payload	Placement	Reaction_wheel_command_distribution
FS: Code_generation	Ada_code_generation	Placement	Reaction_wheels
		Placement	Functional_design
		Placement	Absolute_pointing
		Placement	Sensitivity_analysis
		Placement	Code_generation

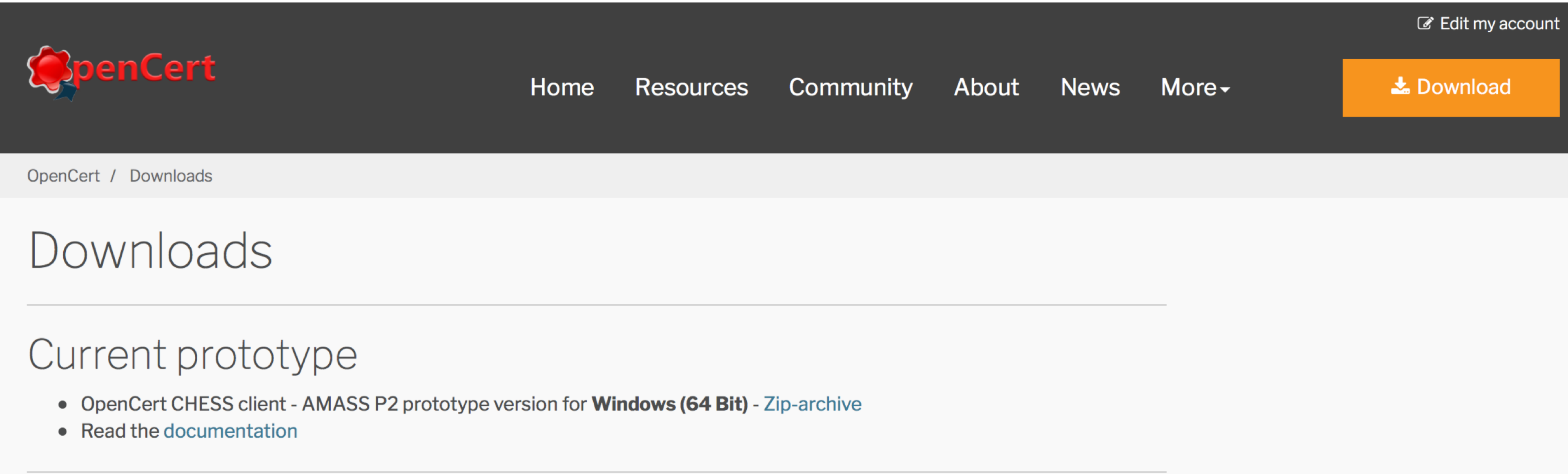
Attitude and Orbit Control Subsystem.uml

- <<CHESS>> <Model> model
  - <<RequirementView>> <Package> modelRequirementView
  - <<SystemView>> <Package> modelSystemView
  - <<ComponentView>> <Package> modelComponentView
  - <<DeploymentView>> <Package> modelDeploymentView
  - <<AnalysisView>> <Package> modelAnalysisView

model.xmi

- Process Package Software Detail Design Method
  - Activity Software Detail Design Method
  - Task Descriptor Object Oriented Design
  - Work Product Descriptor software_design_method
  - Role Descriptor AOCS Designer
  - Role Descriptor AOCS Engineer
  - Role Descriptor Development Team Leader
  - Task Descriptor Functional Design
- Process Package Detailed Design of Real–Time Software

Current implementation



OpenCert / Downloads

Downloads

Current prototype

- OpenCert CHESS client - AMASS P2 prototype version for **Windows (64 Bit)** - Zip-archive
- Read the documentation

https://polarsys.org/opencert/downloads/

Conclusion and Future Work

- The seamless integration between:
  - EPF Composer and BVR Tool
  - CHESS Toolset and BVR Tool
  - OpenCert-Assurance Case Editor and BVR Tool

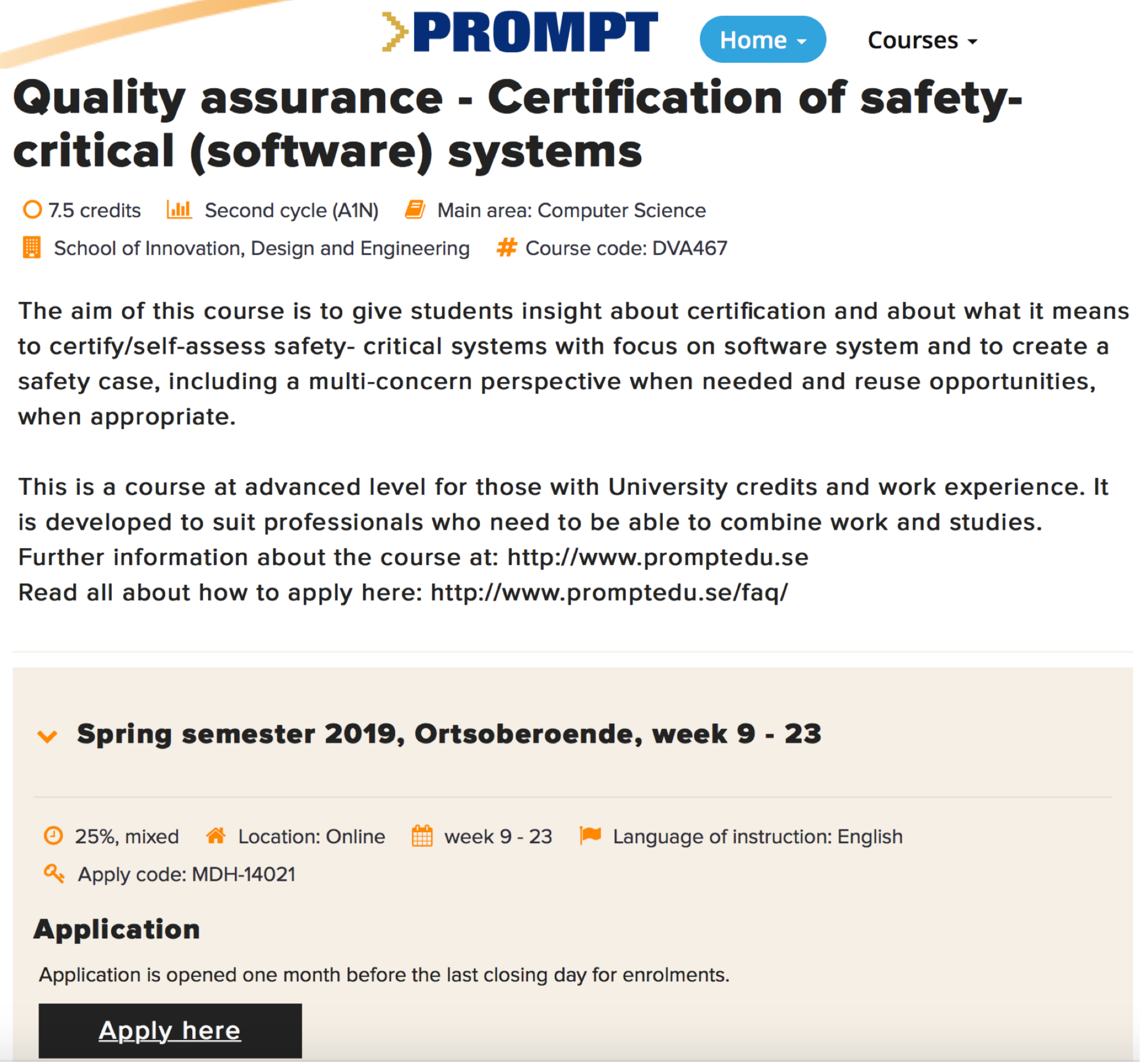
  has been achieved

- → we can support variability management along three dimensions: process, product and assurance case

- Perform extensive validation embracing various product artifacts

MÄLARDALEN UNIVERSITY SWEDEN

AMASS



Quality assurance - Certification of safety-critical (software) systems

○ 7.5 credits Second cycle (A1N) Main area: Computer Science

School of Innovation, Design and Engineering # Course code: DVA467

The aim of this course is to give students insight about certification and about what it means to certify/self-assess safety- critical systems with focus on software system and to create a safety case, including a multi-concern perspective when needed and reuse opportunities, when appropriate.

This is a course at advanced level for those with University credits and work experience. It is developed to suit professionals who need to be able to combine work and studies.
Further information about the course at: http://www.promptedu.se
Read all about how to apply here: http://www.promptedu.se/faq/

Spring semester 2019, Ortsoberoende, week 9 - 23

25%, mixed Location: Online week 9 - 23 Language of instruction: English

Apply code: MDH-14021

Application

Application is opened one month before the last closing day for enrolments.

Apply here

https://www.mdh.se/utbildning/kurser?kod=DVA467&l=en_UK