Vincenzo De Simone

Tutor: Anna Rita Fasolino

XXX Cycle - II year presentation

Supporting the management and improvement of Industrial Software Processes

Research Context

Software crucial in different domains of our everyday life (business, automotive, aerospace, railway, education, etc.)

Software Process defines how software is developed

- ❑ Roles, Activities, Resources

Software Process Improvement (SPI) represents initiatives to reduce *complexity* and *costs* of software process and to improve *software quality*

Motivations for SPI

- ❑ **Compliance** with regulations and quality standard
- ❑ **Handling complexity and variability** of produced software
- ❑ **Coordination of** involved people, resources and **tools**

Compliance with regulations and standards

A Gap Analysis Approach

Objective: support the introduction of regulations or standards requested for software development (e.g. ***ISO/IEC 61508, ISO 26262, SPICE, etc.***) by identifying their requirements the company is not able to carry out

Methodology: questionnaire-based approach exploiting Model Driven Engineering (MDE)

Step 1: Model the standard and the questionnaire

- ✓ definition of UML profiles for generic standards and gap analysis questionnaires

Step 2: Automatically generate a tool supporting the questionnaire completion

- ✓ transformations toward Application Lifecycle Management systems

Step 3: Gap Analysis execution and evaluation exploiting the automatically generated tool

Case Study

Introduction of *ISO 26262 standard* for software development process in the ***Automotive Domain***

Evaluation of the **feasibility** of the proposed approach:

- ➢ It allowed to identify the **gaps** and to plan the needed **improvements**
- ➢ *acceptable* ***application costs***

Handling complexity and variability

Software Product Lines (SPL) Methodology

Objective: propose a solution to cope with the high variability of software systems to be produced

Definition of a methodology to develop a ***SPL infrastructure*** able to generate software architectural models from specification documents. The methodology relies on:

1. ***Definition of a Product Line Architecture*** representing the design common to all products and their possible points of variability
2. ***Definition of the transformation rules*** able to produce an architectural model specific for a given product according to its Features Profile (specifies its variation points and how to configure them)

The obtained SPL infrastructure is exploited to automatically produce specific architectural models tailored to the considered software product leading to cost savings.

Application in real industrial scenarios

The approach was applied in the context of **Instrument Panel Cluster (IPC)** and **Infotainment** embedded Software Development processes → **reduction of implementation costs** for each product from several weeks to days

Coordination of tools

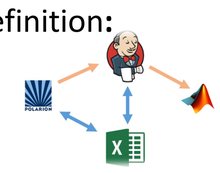
Process automation with tools Integration

Software process requires the coordination of different tasks

- ➢ Manually executed/Time consuming
- ➢ Error-prone
- ➢ Requires different tools

A solution to reduce application time and errors and to improve software process quality attributes is **automation exploiting the integration of the tools used in the process.** The integration requires the definition:

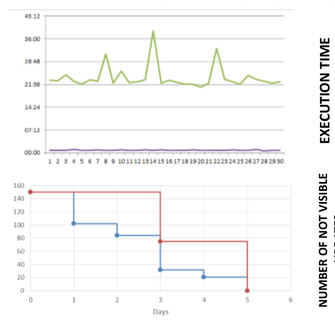
- ➢ call connectors
- ➢ data exchange connectors



Experiment and case study

Application of the approach in Software Testing Processes

- ➢ Evaluation of quality attributes in **comparative studies** (**Manual** vs **Automatic** software process)
- ➢ **semi-structured interviews** with people involved in the software process



Obtained Results:

- ✓ Costs and editing errors reduction
- ✓ Process visibility improvement

Collaborations: APP4Safety Project

METODOLOGIE E TECNOLOGIE INNOVATIVE PER UN APPROCCIO INTEGRATO ALLA SICUREZZA DEL VEICOLO

FCA FIAT CHRYSLER AUTOMOBILES

DIETI UNINA

CeR ICT

Research Group:

REVERSE DIETI UNINA

vincenzo.desimone2@unina.it

Future Work

1. Definition of a methodology for the automatic extraction and documentation of enacted software processes
2. Empirical Evaluation of the quality of the produced documentation with industrial case studies