Vincenzo De Simone

Tutor: Anna Rita Fasolino

XXX Cycle – 3rd year presentation

Assessing and Improving Industrial Software Processes

Università degli Studi di Napoli
Federico II

Background

- **Graduation**: MSc in Computer Engineering at the University of Naples *“Federico II”*
- **Research**: REvERSE Group
- **Fellowship**: P.O.R. F.S.E. grant
- **Research field**: Software Process and Reverse Engineering

Credits Summary

	Credits year 1						Credits year 2						Credits year 3						
	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	
	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	bimonth	**Total**
Modules		5	3	3	9	0	0	3	3	8	0	0	0	0	4	0	0	0	**38**
Seminars	1,5	1,2	2,5	4,8	0	0	3,6	0,8	0,4	2,2	0	0,5	0	0,6	0	3	2	0	**23**
Research	8,5	3,8	4,5	2,2	4	10	6,4	6,2	6,6	1,8	8	9,5	10	9,4	6	7	8	10	**122**
	10	**10**	**10**	**10**	**13**	**10**	**10**	**10**	**10**	**12**	**8**	**10**	**10**	**10**	**10**	**10**	**10**	**10**	**183**

Context

Industrial Software Processes

Objective

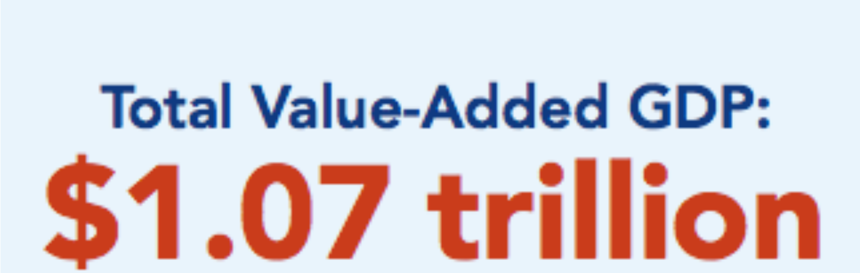
Supporting the assessment and improvement of Industrial Software Processes by:

- proposing novel approaches
- designing and developing innovative tools

Motivations 1/2

Software has a great impact on our lives *

Economic Impact



Total Value-Added GDP:
$1.07 trillion

Social Impact

The apps we use every day are software

Data is driven by software

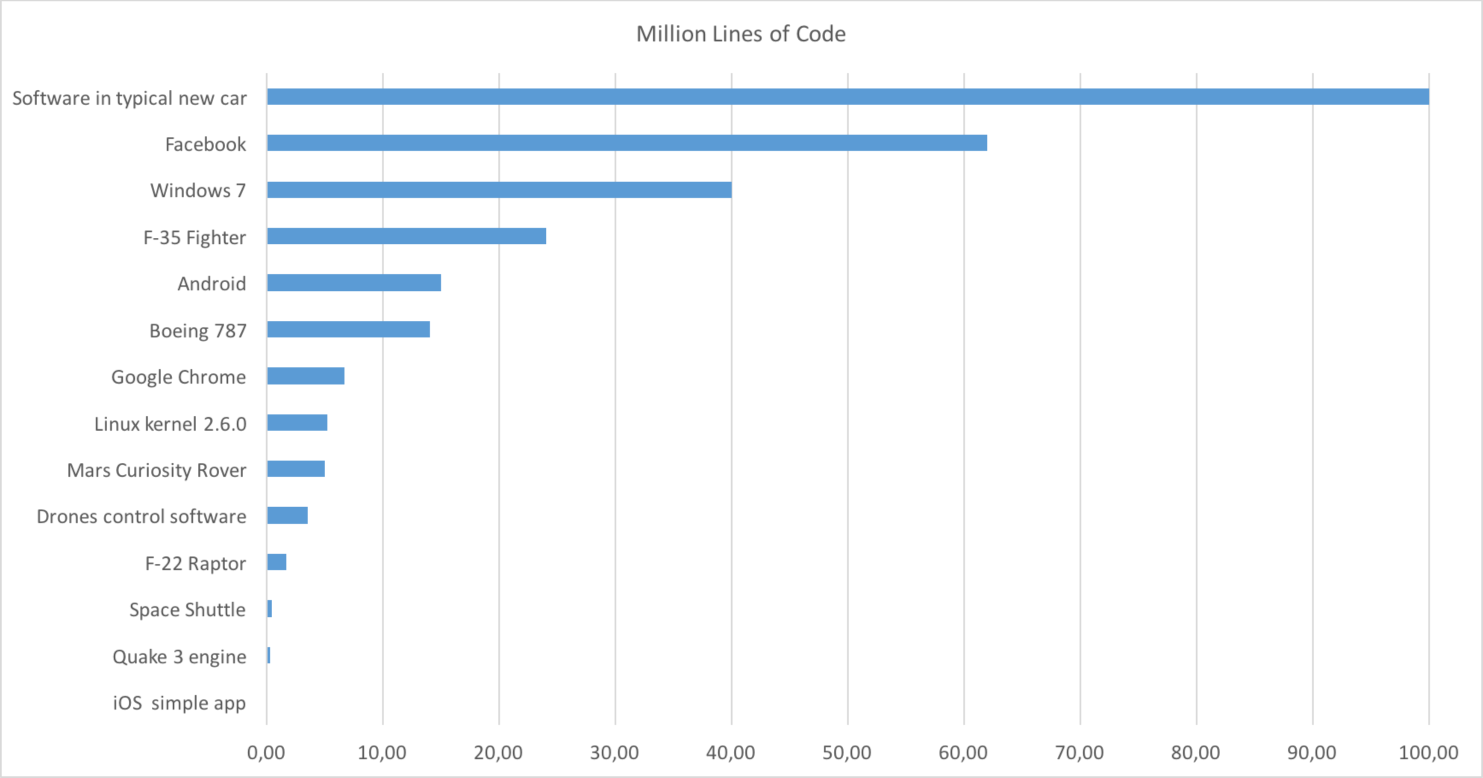
Cloud and IOT are enabled by software

Industry solutions are powered by software

* www.bsa.org/softwareimpact

Motivations 2/2

- Software complexity is growing



Excerpt of http://www.informationisbeautiful.net/visualizations/million-lines-of-code/

The Research Topic

- ➢ Software crucial part of systems in different domains
- ➢ Strict time-to-market
- ➢ Required compliance with Standard regulations and Software Quality Standards

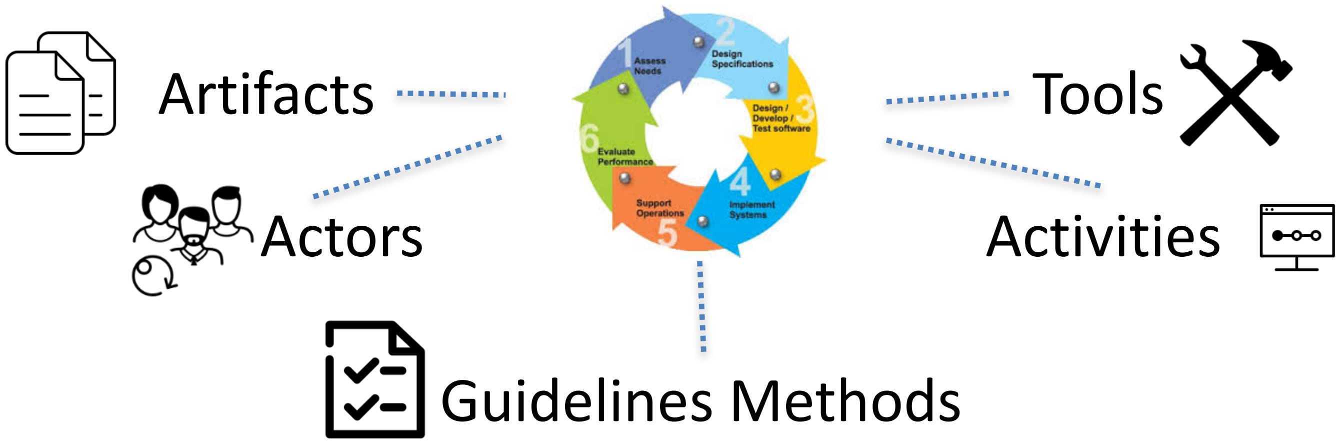
➔

Software Process is a strategic factor for addressing the software problems of costs, quality, and scheduling

Software Process

Complex phenomenon for developing and maintaining a software product or a software-based product [1]

SOFTWARE PROCESS



Artifacts

Actors

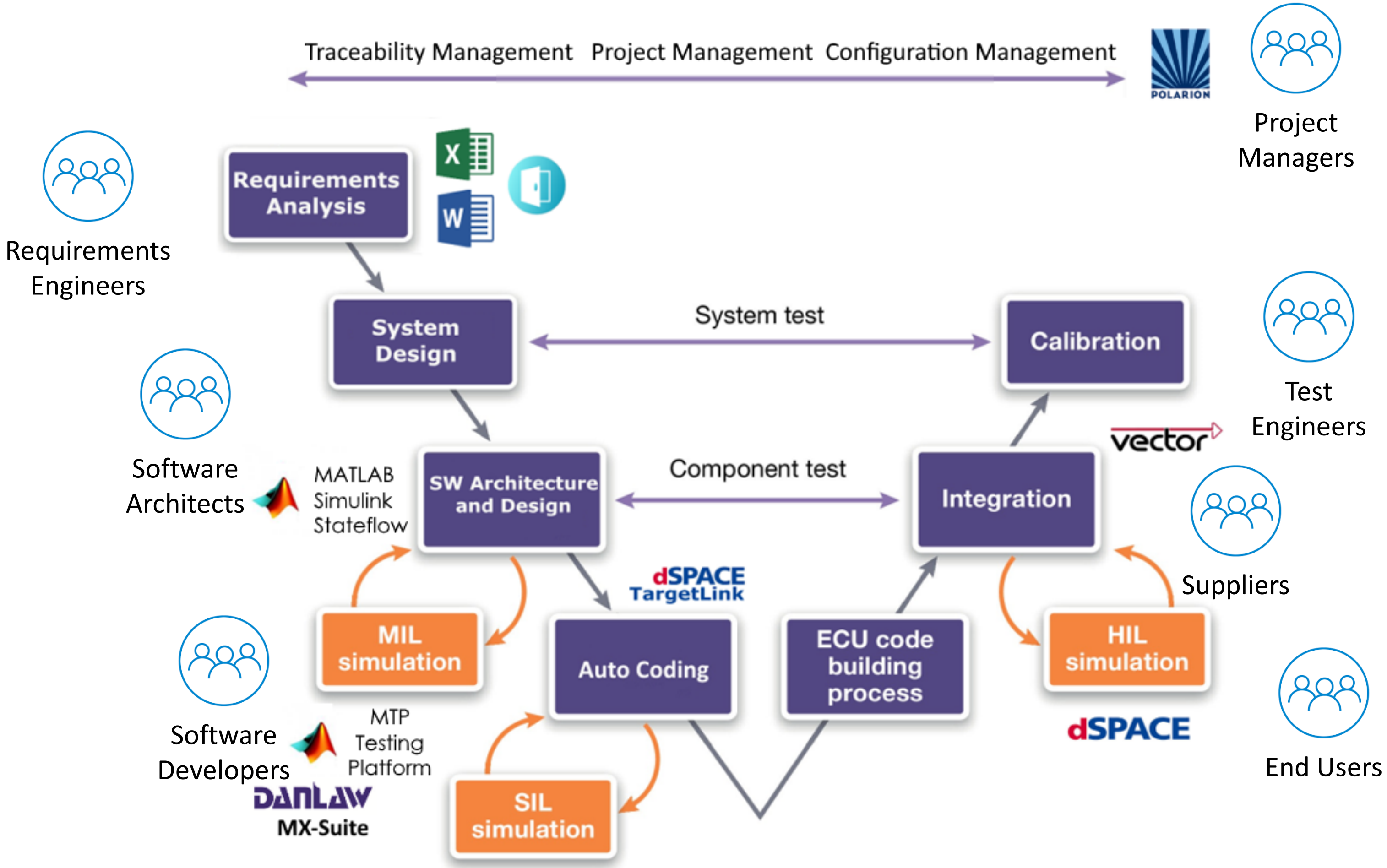
Tools

Activities

Guidelines Methods

[1] Alfonso Fuggetta, Elisabetta Di Nitto:Software process. FOSE 2014

Industrial Software Process



Industrial Software Process Issues 1/2

- Software companies are asked to assess if their processes comply with Standards or Quality Evaluation Frameworks to be competitive on their markets and to guarantee the quality of their products.
  - Complex process to be effectively executed.
- Difficult to manage traceability among the software artifacts involved in development of software process, due to their complexity and their size.
  - This has a negative impact on the quality of the software processes and of the software products.

Industrial Software Process Issues 2/2

- Artifacts are complex, not documented resulting hard to be comprehended and maintained.

Need for solutions for **assessing** software processes and **improving** them by effectively managing the involved software artifacts

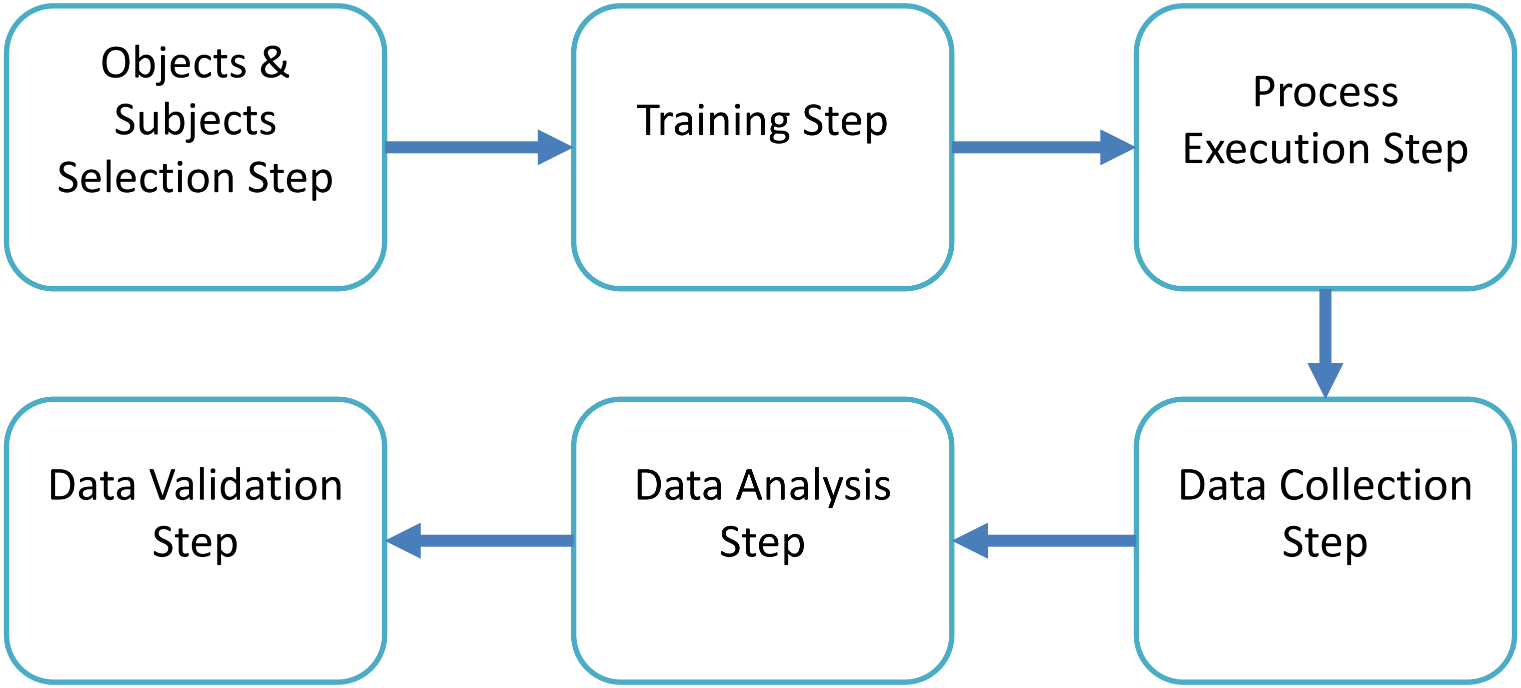
Contributions

- An approach for supporting Software Process Assessment through Gap Analysis with respect to Standards and Quality Evaluation Frameworks;
- An approach for the automatic management of traceability links among the artifacts involved in software processes;
- A reverse engineering process and a tool for supporting the comprehension of spreadsheet based artifacts involved in software processes;
- The evaluation of the proposed approaches in real industrial contexts exploiting case study research method

Case Study Research

- Empirical research method best suited to be applied for problems that need to be studied in context.
    - Phenomena under study cannot be separated from context. Effects can be wide-ranging.
        - How and why questions
    - To gain a deep understanding of a phenomenon
        - Exploratory, explanatory or validation purposes

Experimental Procedure



Experimental Context

The case studies were carried out in the automotive domain, in the context of the APPS4Safety Research Project.

Software Process Assessment

Software Process Assessment

- Standard of Quality Evaluation Frameworks
  - **defines requirements** to be met for reaching a certain level of quality of the software process and consequently of the software products. Examples
    - *CMMI* (Capability Maturity Model Integration)
    - *SPICE* (Software Process Improvement and Capability Determination)
    - *ISO/IEC 61508 (*Functional Safety of Safety-related systems)
    - *ISO 26262* (Road vehicles – Functional safety)
- There is the need for companies to **assess** their processes against these Standards

Gap Analysis for Software Process Assessment

- **Gap Analysis** allows to determine how far is the company from implementing the requirements reported by a Standard
  - ***As is*** versus ***To be*** situation
  - usually carried out through questionnaire based processes
    - execution affected by several issues

Gap Analysis
Issues

- Survey conducted in industrial settings for understanding:
  - how gap analysis processes are actually executed
    - Gap Analysis Questionnaire Design, Questionnaire Completion, Questionnaire Analysis
  - the involved artifacts
    - Document of the Standard, Questionnaires, Assignment Forms, Reports
  - the required actors and their roles
    - Questionnaire Designer, Respondents, Responsible, Reviewer, Internal Assessor
  - the exploited tools
    - Spreadsheets, Word Processors
  - the main issues
    - lack of proper methodology, tool support, management of artifacts lifecycle

Gap Analysis
Issues

Need for techniques and tools supporting:

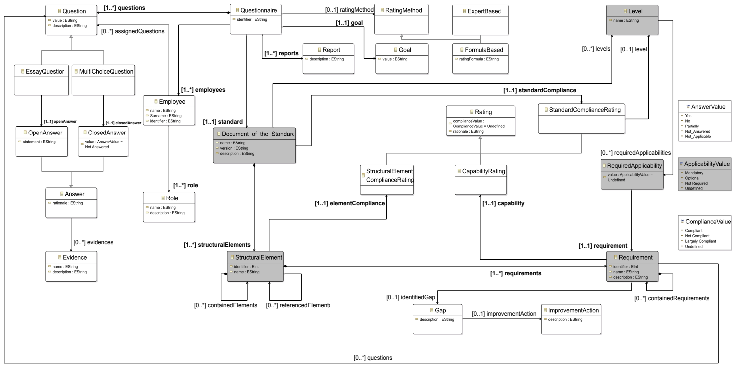
- the Gap Analysis process design
- the Gap Analysis process execution

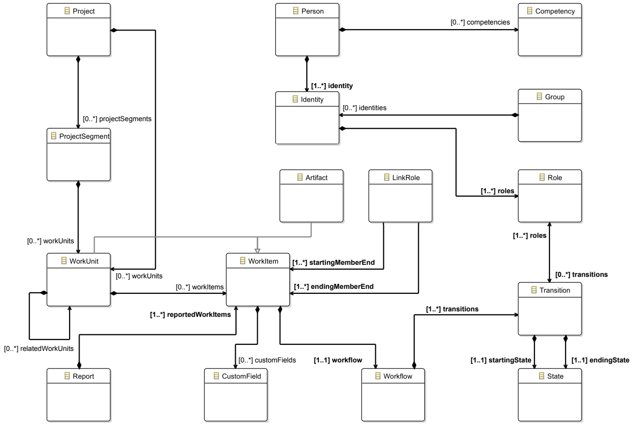
Proposed Solution for Software Process Assessment

- Approach for supporting Gap Analysis Processes execution exploiting Application Lifecycle Management (ALM) features that can be tailored according to the company needs and the considered Standard
- A Model Driven Engineering approach and tool, GADGET, for designing a Gap Analysis Process and automatically developing a software infrastructure supporting its execution

The MDE Approach

- Realization of novel conceptual models abstracting the characteristics of:
  - Gap Analysis Processes
  - Application Lifecycle Management

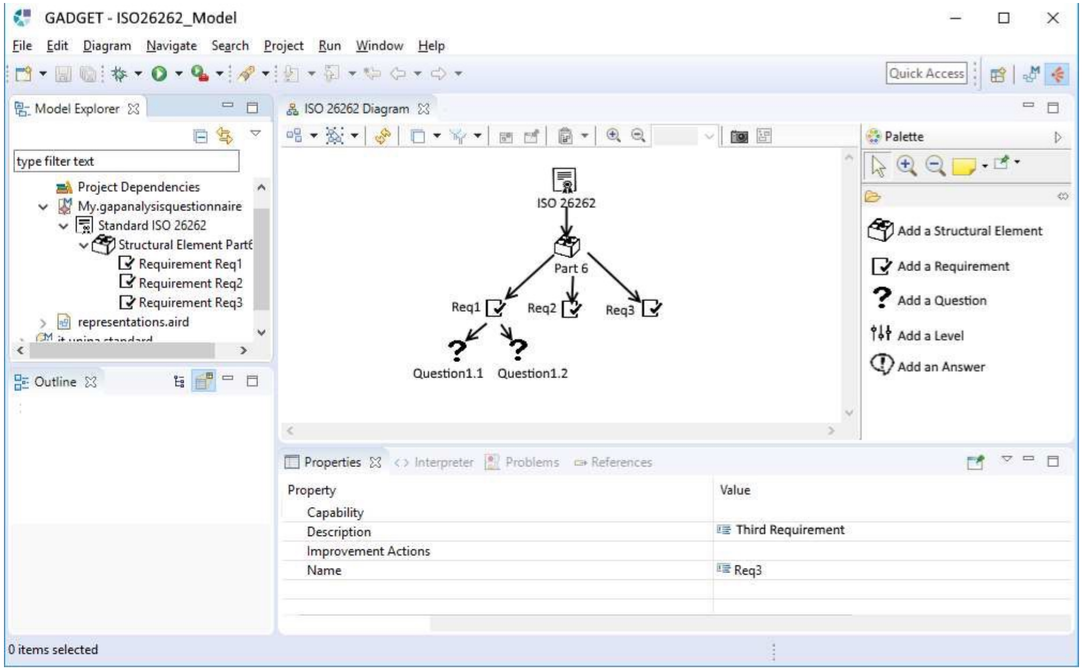




Vincenzo De Simone

The GADGET Tool

- Realization of a Graphical Workbench[2] for easing the design, in a systematic way, of Gap Analysis Processes

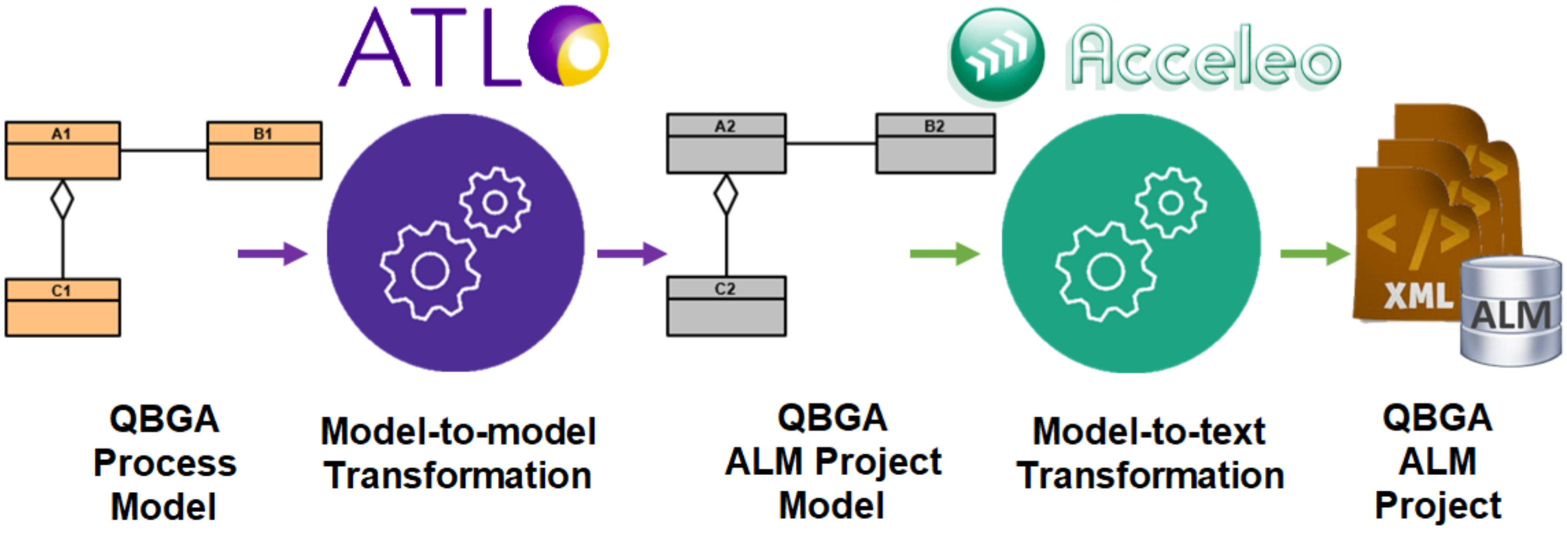


Property	Value
Capability	
Description	Third Requirement
Improvement Actions	
Name	Req3

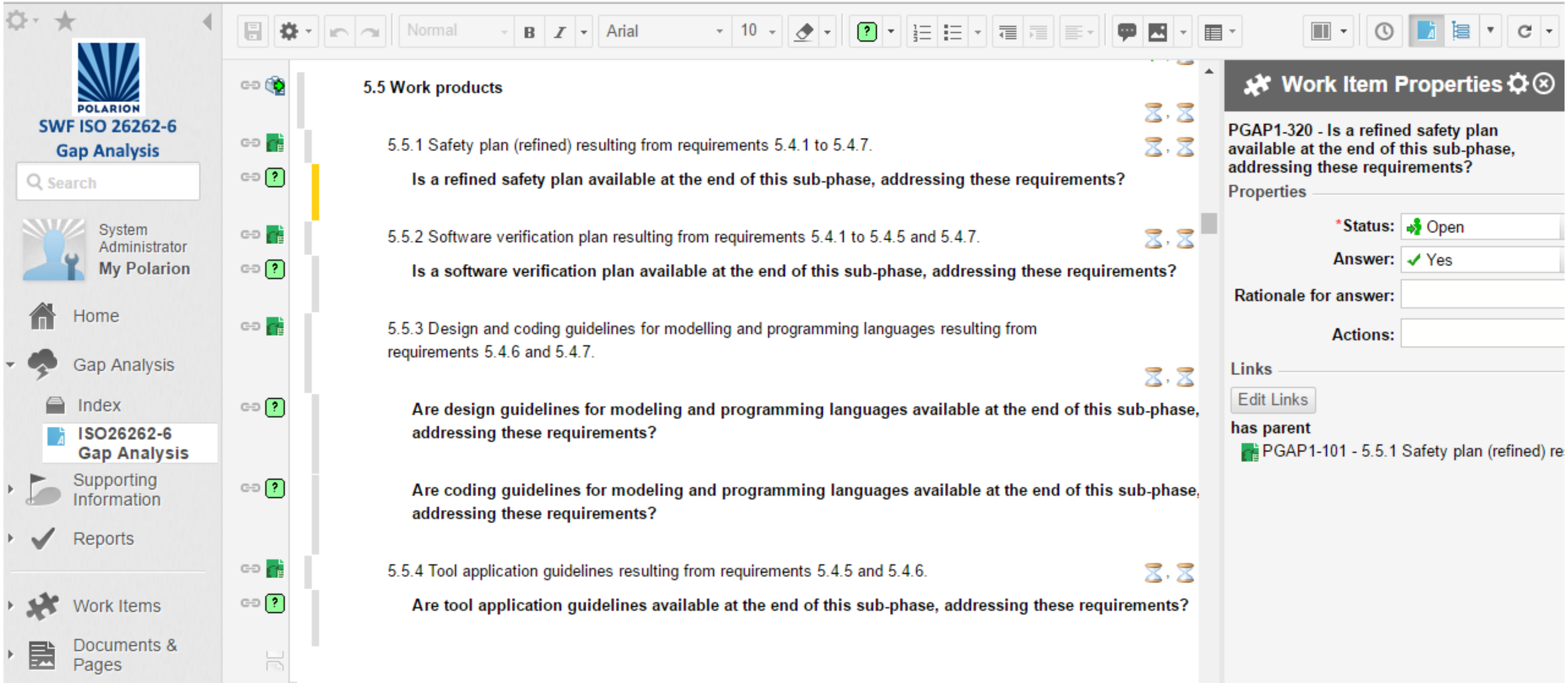
[2] based on Eclipse SIRIUS https://www.eclipse.org/sirius/

The Automatic Transformation Process

- Realization of an automatic transformation process for realizing an ALM project for the execution of the designed Gap Analysis Process.



The Gap Analysis ALM Project



5.5 Work products

5.5.1 Safety plan (refined) resulting from requirements 5.4.1 to 5.4.7.

Is a refined safety plan available at the end of this sub-phase, addressing these requirements?

5.5.2 Software verification plan resulting from requirements 5.4.1 to 5.4.5 and 5.4.7.

Is a software verification plan available at the end of this sub-phase, addressing these requirements?

5.5.3 Design and coding guidelines for modelling and programming languages resulting from requirements 5.4.6 and 5.4.7.

Are design guidelines for modeling and programming languages available at the end of this sub-phase, addressing these requirements?

Are coding guidelines for modeling and programming languages available at the end of this sub-phase, addressing these requirements?

5.5.4 Tool application guidelines resulting from requirements 5.4.5 and 5.4.6.

Are tool application guidelines available at the end of this sub-phase, addressing these requirements?

PGAP1-320 - Is a refined safety plan available at the end of this sub-phase, addressing these requirements?

Properties

*Status:	Open
Answer:	✓ Yes
Rationale for answer:	
Actions:	

Links

has parent

PGAP1-101 - 5.5.1 Safety plan (refined) re

Software Process Assessment Case Study 1/2

Goal: Evaluate the **feasibility** of the approach for supporting Software Process Assessment

- **$RQ_{1.1}$** *How does the GADGET tool support the design and development of ALM-based tools for aiding the Gap Analysis Processes?*
- **$RQ_{1.2}$** *How does ALM-based tools affect the Gap Analysis Processes execution?*

Objects: ***2 Gap Analysis processes*** with respect to the part 6 and part 9 of the Functional Safety Standard ISO 26262

Subjects: ***12 Employees*** of an Automotive Company

Software Process Assessment Case Study 2/2

RESULTS

$RQ_{1.1}$ *GADGET well supported both the Questionnaire developers in carrying out their design and development tasks.*

$RQ_{1.2}$ *The adoption of ALM for supporting Gap Analysis Processes positively affected their execution, improving the process visibility, acceptability and supportability from the point of view of all the involved actors.*

Tool Integration Approach

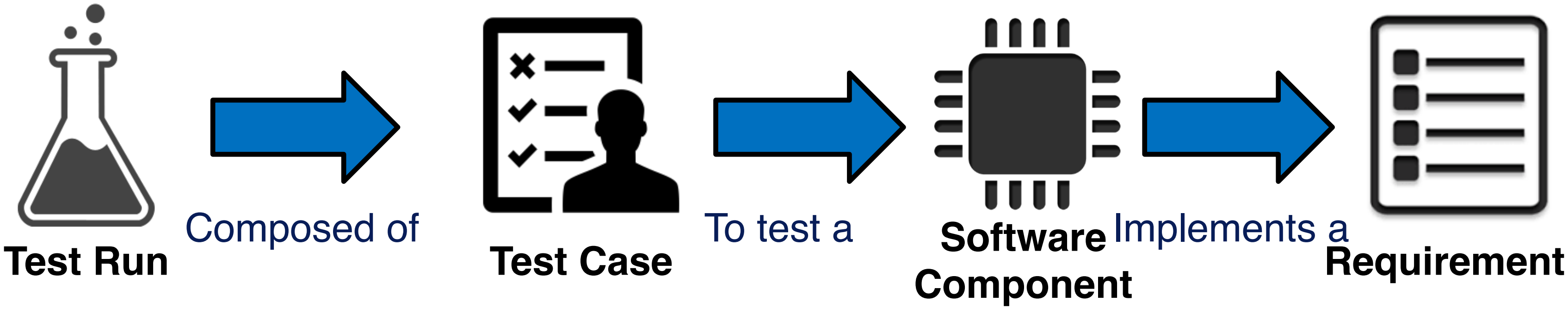
Tool Integration Approach

The lack of tool integration negatively affects software process quality in terms of:

- **Visibility** Absence of accurate, reliable and timely information. Difficult to follow the process progress and results.
- **Rapidity** High software process execution time due to long time needed to exchange data between the different tools
- **Ineffective Traceability Management** Manual creation of the needed traceability links, that was time consuming and error prone
- **Acceptability** actors involved in the process have to use many different tools at the same time and execute tedious, time consuming and error prone manual tasks

Software Traceability

- creation and the use of links (or connections) between different kinds of software artifacts such as requirements, models, source code, test cases, or test results. It is required by **Quality Evaluation Frameworks** and **Safety Standards**.



Current Approaches for Traceability Management

- Traceability management is not well supported in the practice
  - Use of spreadsheet for defining traceability links (time consuming, error-prone)
- Different approaches proposed in the literature for recovering possible trace links a posteriori
  - No approach have been proposed to automatically create them when they are established

The proposed Tool Integration Approach 1/2

The Tool Integration Approach aims at:

- automating repetitive and time consuming tasks according to a defined workflow
- enabling the integration among the software process tools
- managing the involved artifacts and for the automatic creation of trace links

It was realized by adapting Commercial Off-The-Shelf solutions to exploit already provided functionality and improve maintainability.

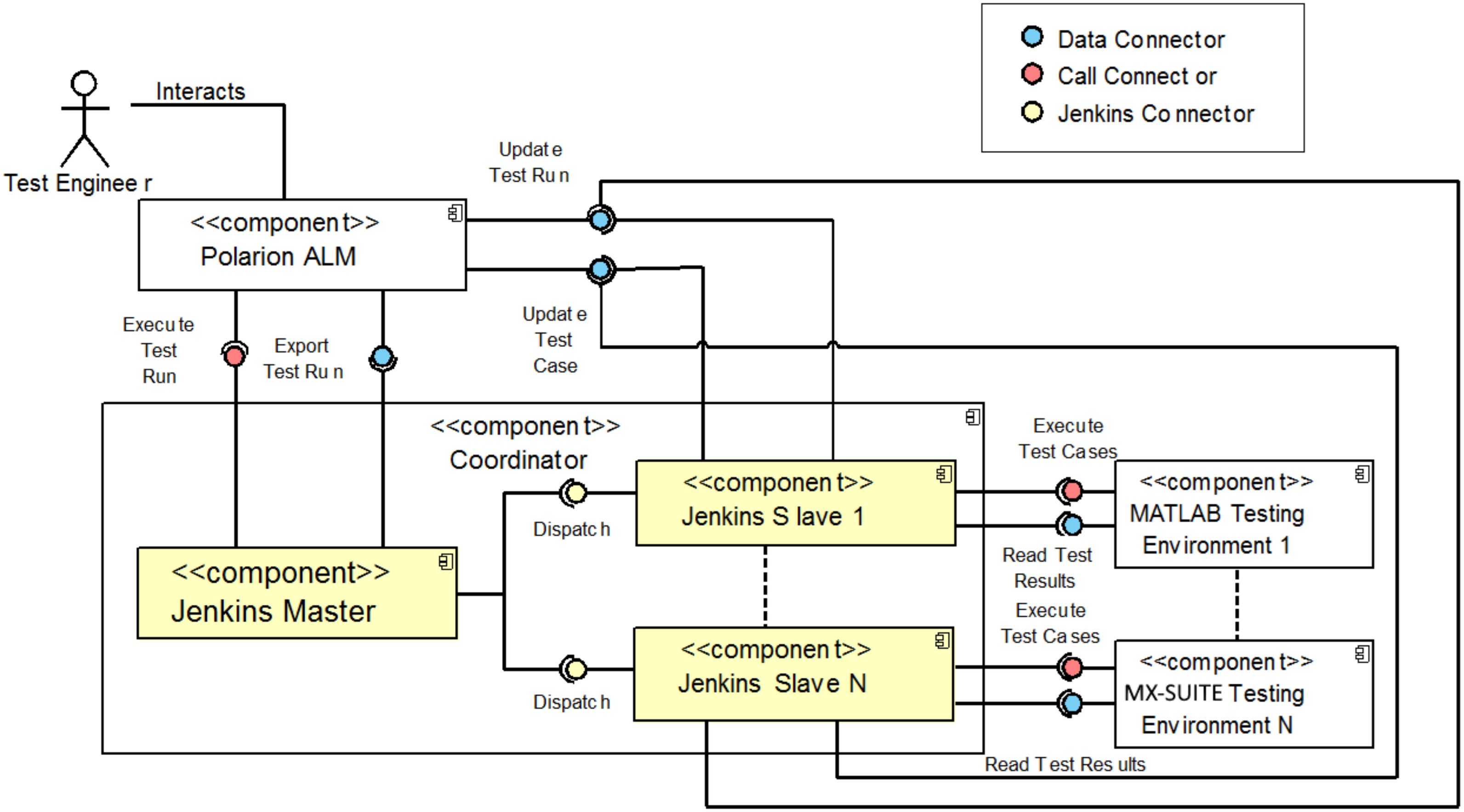
  – It exploits ALM and Continuous Integration Systems.

The proposed Tool Integration Approach 2/2

To realize the proposed approach requires reverse engineering of

- process workflow
  - for supporting its automation
- data models of the involved artifacts and their links
  - to enable their interoperability
- the tools involved in the process
  - to wrap of the features of legacy tools for make them expose callable APIs

The proposed Tool Integration Architecture


Data Connector
Call Connector
Jenkins Connector
Test Engineer
Interacts
Update Test Run
<<component>> Polarion ALM
Execute Test Run
Export Test Run
Update Test Case
<<component>> Coordinator
<<component>> Jenkins Slave 1
<<component>> Jenkins Master
Dispatch
<<component>> Jenkins Slave N
Dispatch
Execute Test Cases
<<component>> MATLAB Testing Environment 1
Read Test Results
Execute Test Cases
<<component>> MX-SUITE Testing Environment N
Read Test Results

Tool Integration Approach
Case Study 1/2

Goal: Evaluate the **effects** of the Tool Integration Architecture on the quality of real software processes against the usual company practices

- **$RQ_{2.1}$** *How does the adoption of the proposed solution affect the process rapidity?*
- **$RQ_{2.2}$** *How does the adoption of the proposed solution influence the effectiveness of the traceability links management related to the process?*
- **$RQ_{2.3}$** *How does the adoption of the proposed solution impact on the process visibility from the point of view of the Project Manager?*
- **$RQ_{2.4}$** *How does the adoption of the proposed solution impact the process acceptability from the point of view of the Test Engineers?*

Tool Integration Approach
Case Study 2/2

- **Objects:**
  - Software processes for the testing of **3 different Software Components**
  - **(Speedometer, Tachometer, Trip)**
- **Subjects:**
  - **1 Project Manager** and **3 Test Engineers** of an Automotive Company

Case Study Results 1/4

- **Process Rapidity**

$$Speedup\ Percentage = \frac{T_{original} - T_{new}}{T_{original}} * 100$$

SWC	MIL	CTR	STC	LTC	ITCR
Speedometer	7%	0%	73%	0%	79%
Tachometer	9%	0.1%	81%	0%	90%
Trip	14%	0.1%	91%	0.3%	96%

MIL = Entire MIL Testing Process
CTR = Create Test Run
STC = Select Test Cases
LTC = Launch Test Cases
ITCR = Import Test Cases Result

Case Study Results 2/4

- **Effectiveness Traceability Management**

ITLRP = *ITL Reduction Percentage* = $\frac{ITL_{original} - ITL_{new}}{ITL_{original}} * 100$

MTLRP = *MTL Reduction Percentage* = $\frac{MTL_{original} - MTL_{new}}{MTL_{original}} * 100$

SWC	ITLRP	MTLRP
Speedometer	100%	100%
Tachometer	100%	100%
Trip	100%	100%

Case Study Results 3/4

- **Visibility**
  - Improved from the point of view of the Project Manager
    - He report the possibility to monitor the progress and results of the MIL testing process execution when needed
    - In the past he could obtain the needed information only on defined milestones or by directly querying the involved Test Engineers

Case Study Results 4/4

- **Acceptability**
  - Improved from the point of view of the Test Engineers
    - No need to switch between different tools since the execution of their tasks is carried out through one tool (the ALM). The need for manual interventions was reduced
    - In the past they had to switch between different tools executing tedious manual tasks (e.g. storing of the test results)

Comprehending Spreadsheet based Artifacts

Comprehending Spreadsheet based Artifacts

Spreadsheet based artifacts are widely adopted in different industrial software processes

- for supporting several of its phases, i.e. requirement and test case definition, activity reports, etc.

These artifacts are complex and difficult to be comprehended since they lack of a clear separation between the presentation, business logic and data presentation

- Negative impact on the quality of software processes where they are adopted

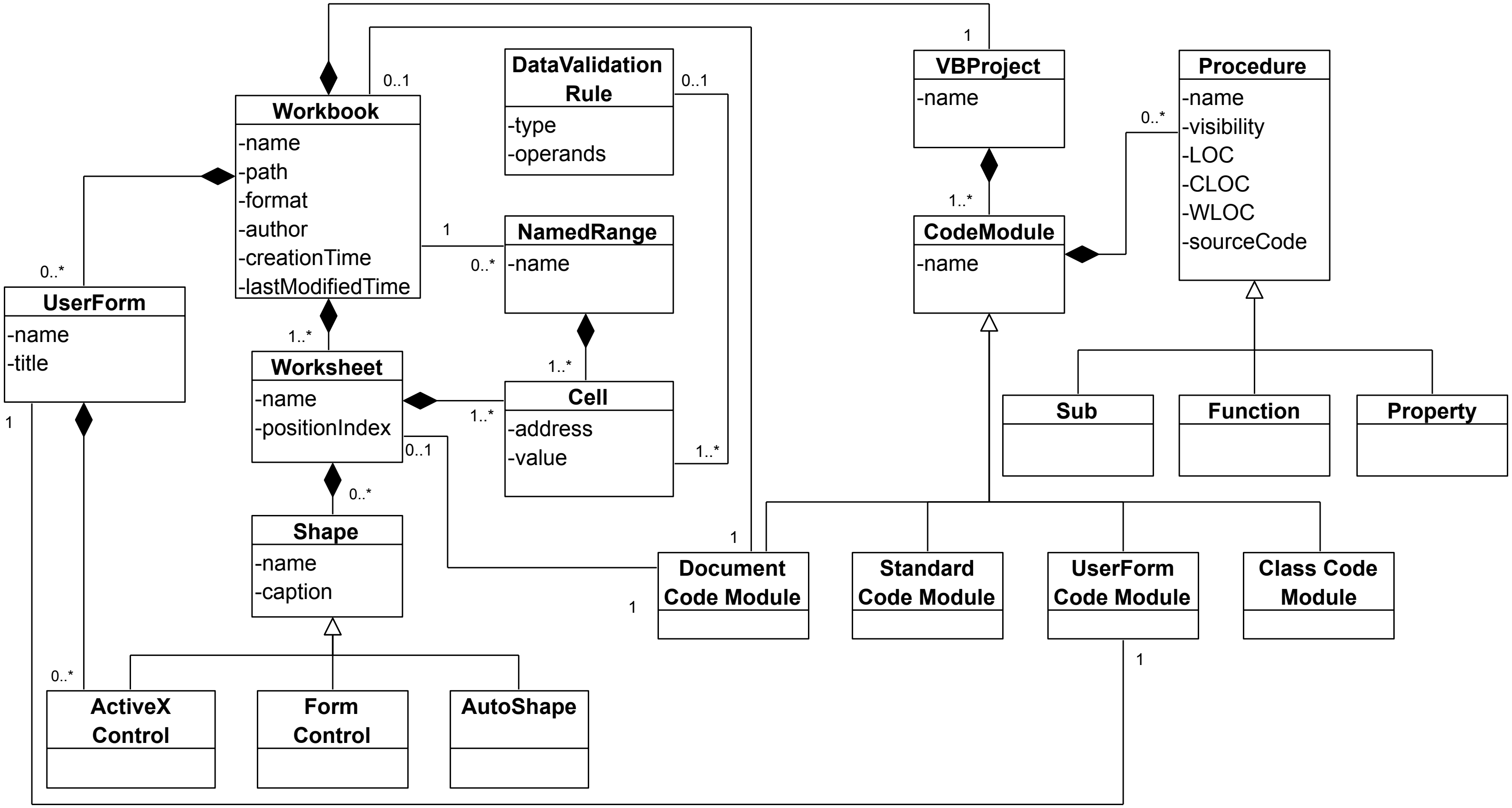
Different approaches already proposed in the literature that did take into account just formulas or their data. They did not consider their business logic and the possible dependencies that they may create.

Proposed Solutions

Definition of a conceptual model of Spreadsheet based artifact and of a reverse engineering process and a tool, named EXACT, for:

- ✓ recovering the data model from the spreadsheet artifacts
- ✓ analyzing their business logic,
- ✓ recovering dependencies among their composing elements
- ✓ providing interactive views supporting their comprehension and maintenance
- ✓ enabling their integration

Conceptual Model of Spreadsheet based Artifacts



The Reverse Engineering Process 1/2

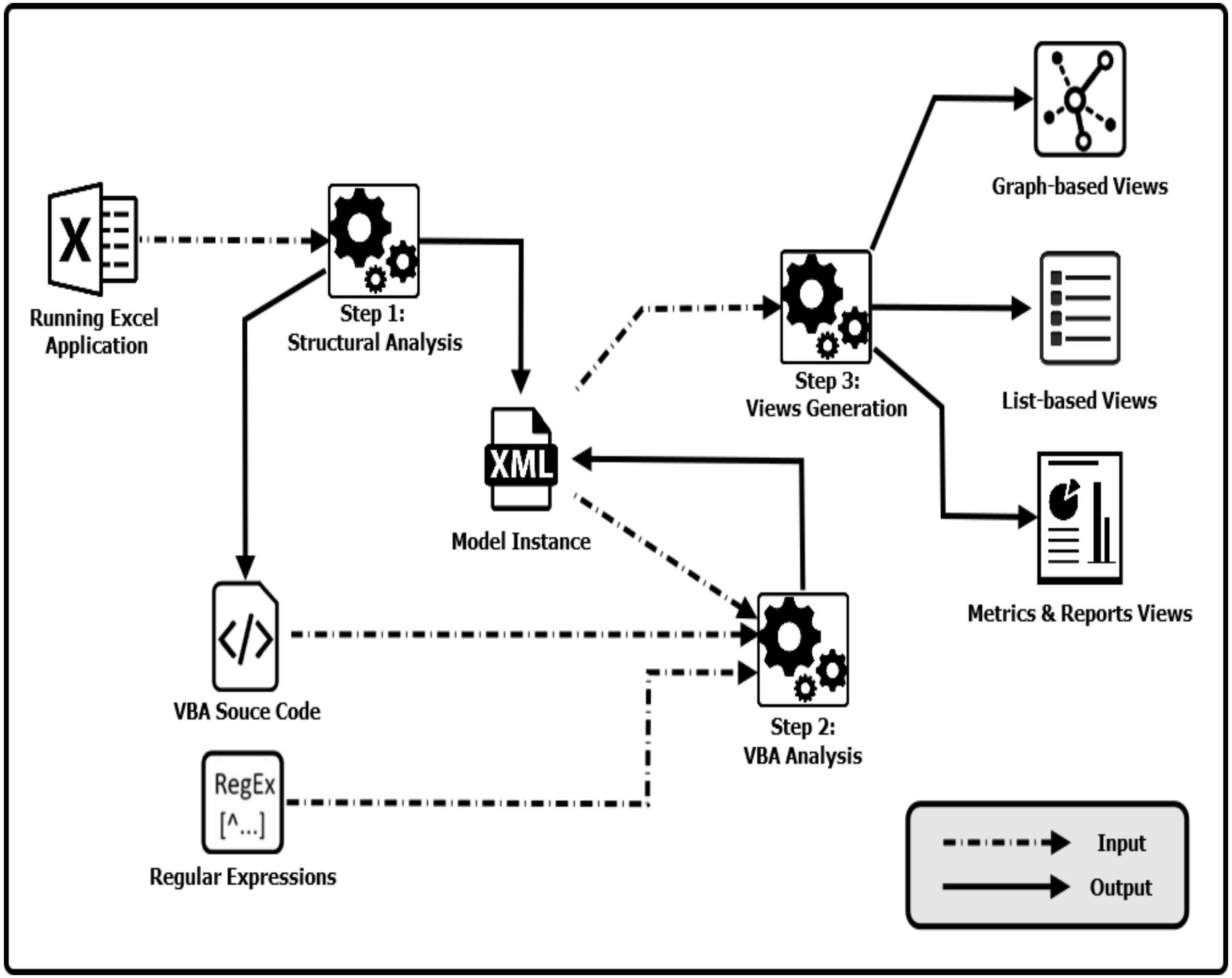
It supports spreadsheet based artifacts comprehension by providing:

- **Analysis features** through static and dynamic analysis for abstracting the conceptual model of the artifact
- **Visualization features** for generating interactive views aiding the users' comprehension

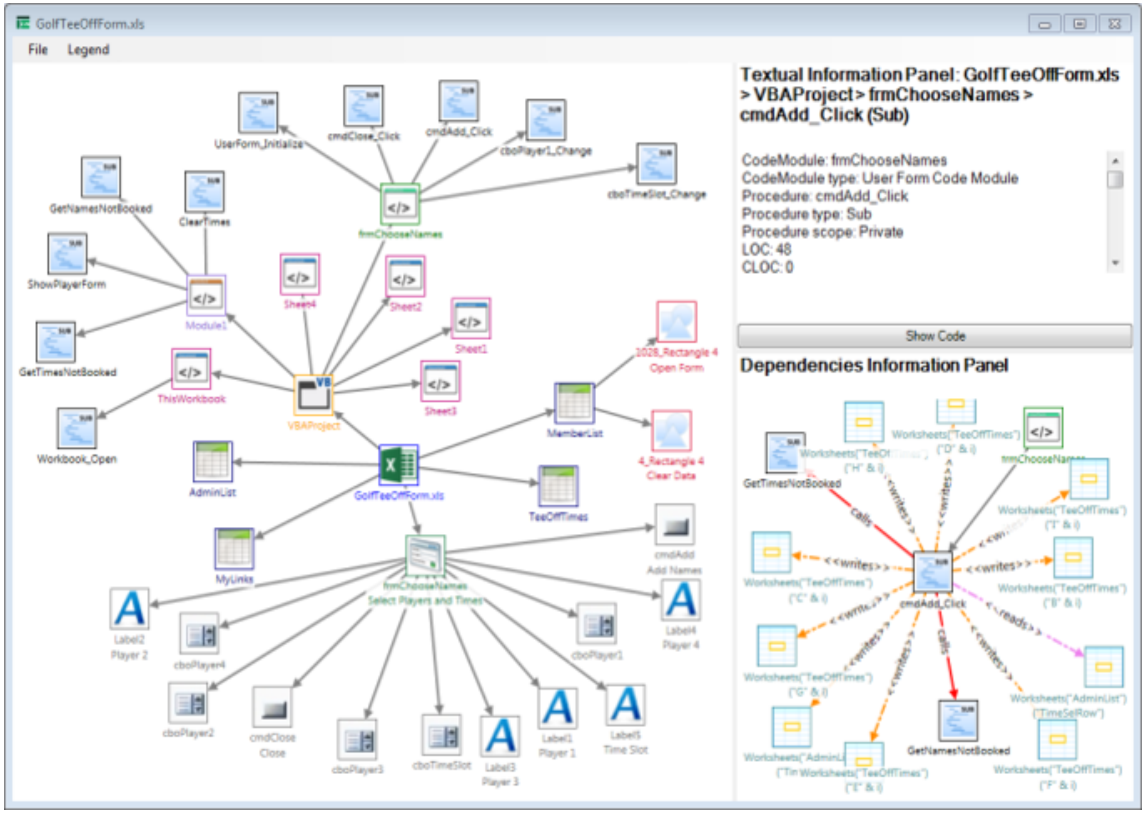
The Reverse Engineering Process 2/2

The reverse engineering process consists of the following steps:

- **Step 1:** Analysis of the application to reconstruct its model
- **Step 2:** Analysis of the VBA code
- **Step 3:** Views Generation



The EXACT Tool



Textual Information Panel: GolfTeeOffForm.xls > VBAProject > frmChooseNames > cmdAdd_Click (Sub)

CodeModule: frmChooseNames
CodeModule type: User Form Code Module
Procedure: cmdAdd_Click
Procedure type: Sub
Procedure scope: Private
LOC: 48
CLOC: 0

Dependencies Information Panel

Visualization:

- Reports the artifact composing elements

Interactive views:

- By clicking on an element, further details are reported along with a view showing its dependencies.

Case Study 1/2

Goal: Evaluate the **validity** of the reverse engineering tool *EXACT* for comprehending and maintaining real spreadsheet based artifacts

- **$RQ_{3.1}$** How does EXACT support professional end users to comprehend VBA-based spreadsheet artifacts?
- **$RQ_{3.2}$** What are the main limitations of EXACT according to the end-users' point of view?

Objects: 3 maintenance projects of different spreadsheet based artifacts corpora

Subjects: 15 Employees of an Automotive Company

Case Study 2/2

Results

RQ$_{3.1}$ *EXACT features helped end users in accomplishing their tasks avoiding them tedious and repetitive tasks. They exploited both systematic and ad-hoc comprehension strategies.*

RQ$_{3.2}$ *The end users lamented a lack of*

- *advanced searching features,*
- *features for the analysis of Charts*
- *clustering mechanisms for improving views readability*

Conclusions

- ❑By exploiting Software engineering methods and enabling technologies I was defined different approaches and tools for assessing and improving Industrial Software Processes.
- ❑The proposed approaches were evaluated with real industrial case studies were they proved their effectiveness.

Future Directions

- ❑Further improvement of the proposed approaches and tools
- ❑Wider evaluation of their validity through case studies involving software processes in different domains, considering a greater number of subjects and objects.
- ❑Integrate the validated approaches in a collaborative and configurable environment, based on MDE and ALM technologies, for supporting Software Process Improvement initiatives

Products

Published Papers 1/2

[C7] Domenico Amalfitano, ***Vincenzo De Simone***, Anna Rita Fasolino, Stefano Scala: **Improving traceability management through tool integration: an experience in the automotive domain.** 10th International Conference on Software and System Process ICSSP 2017: 5-14

[J1] Domenico Amalfitano, ***Vincenzo De Simone***, Anna Rita Fasolino, Porfirio Tramontana: **EXACT: A tool for comprehending VBA-based Excel spreadsheet applications.** Journal of Software: Evolution and Process 28(6): 483-505 (2016). Wiley.

[C6] Domenico Amalfitano, ***Vincenzo De Simone***, Anna Rita Fasolino, Mario Lubrano, Stefano Scala:
Introducing Software Product Lines in Model-Based Design Processes: An Industrial Experience. 13th Working IEEE/IFIP Conference on Software Architecture WICSA 2016: 287-290

[C5] Domenico Amalfitano, Nicola Amatucci, ***Vincenzo De Simone***, Anna Rita Fasolino, Porfirio Tramontana: **Toward Reverse Engineering of VBA Based Excel Spreadsheet Applications.** 2nd International Workshop on Software Engineering Methods in Spreadsheets SEMS@ICSE 2015: 30-31

Published Papers 2/2

[C4] Domenico Amalfitano, ***Vincenzo De Simone***, Anna Rita Fasolino, Vincenzo Riccio: **Comparing Model Coverage and Code Coverage in Model Driven Testing: An Exploratory Study.** 30th IEEE/ACM International Conference on Automated Software Engineering Workshops 2015: 70-73

[C3] Domenico Amalfitano, Anna Rita Fasolino, Porfirio Tramontana, ***Vincenzo De Simone***, Giancarlo Di Mare, Stefano Scala: **A Reverse Engineering Process for Inferring Data Models from Spreadsheet-based Information Systems: An Automotive Industrial Experience.** Communications in Computer and Information Science 178 - DATA 2014 Revised Selected Papers: 136-153. Springer.

[C2] Domenico Amalfitano, Anna Rita Fasolino, Porfirio Tramontana, ***Vincenzo De Simone***, Giancarlo Di Mare, Stefano Scala: **Information Extraction from Legacy Spreadsheet-based Information System - An Experience in the Automotive Context.** 3rd International Conference on Data Management Technologies and Applications DATA 2014 : 389-398

[C1] Domenico Amalfitano, Anna Rita Fasolino, Valerio Maggio, Porfirio Tramontana, ***Vincenzo De Simone***: **Reverse Engineering of Data Models from Legacy Spreadsheets-Based Systems: An Industrial Case Study.** 22nd Italian Symposium on Advanced Database Systems SEBD 2014: 123-130

Accepted Papers

[C8] Domenico Amalfitano, Nicola Amatucci, ***Vincenzo De Simone***, Vincenzo Riccio, Anna Rita Fasolino: **Towards a Thing-In-the-Loop approach for the verification and validation of IoT systems.** To appear in the Proceedings of 1st ACM Workshop on the Internet of Safe Things

Submitted Papers

[S1] Domenico Amalfitano, ***Vincenzo De Simone***, Stefano Scala, Anna Rita Fasolino: **Using Application Lifecycle Management and Model Driven Engineering for supporting Questionnaire-Based Gap Analysis Processes.** Submitted to the Information and Software Technology Journal. Elsevier.

[S2] Domenico Amalfitano, Nicola Amatucci, ***Vincenzo De Simone***, Vincenzo Riccio, Anna Rita Fasolino : **Is This the Lifecycle We Really Want? An Automated Black-Box Testing Approach for Android Activities.** Submitted to the 11th IEEE Conference on Software Testing, Validation and Verification

Papers In Preparation

[P1] Domenico Amalfitano, ***Vincenzo De Simone***, Anna Rita Fasolino: **Automatic Documentation of Software Processes enacted through Application Lifecycle Management Systems .**

[P2] Domenico Amalfitano, Nicola Amatucci, ***Vincenzo De Simone***, Vincenzo Riccio, Anna Rita Fasolino: **Combining automatic exploration of mobile apps behavior with capture and replay through Machine Learning.**

[P3] Domenico Amalfitano, Anna Rita Fasolino, ***Vincenzo De Simone***, Stefano Scala: **Extended version of Improving Traceability Management through Tool Integration: an Experience in the Automotive Domain.** To be submitted at Journal of Software: Evolution and Process - Special Issue: Revised Selected Papers of ICSSP 2017. Wiley.

Thanks for your attention

Questions? Feedbacks?
Possible Collaborations?

Further Information:

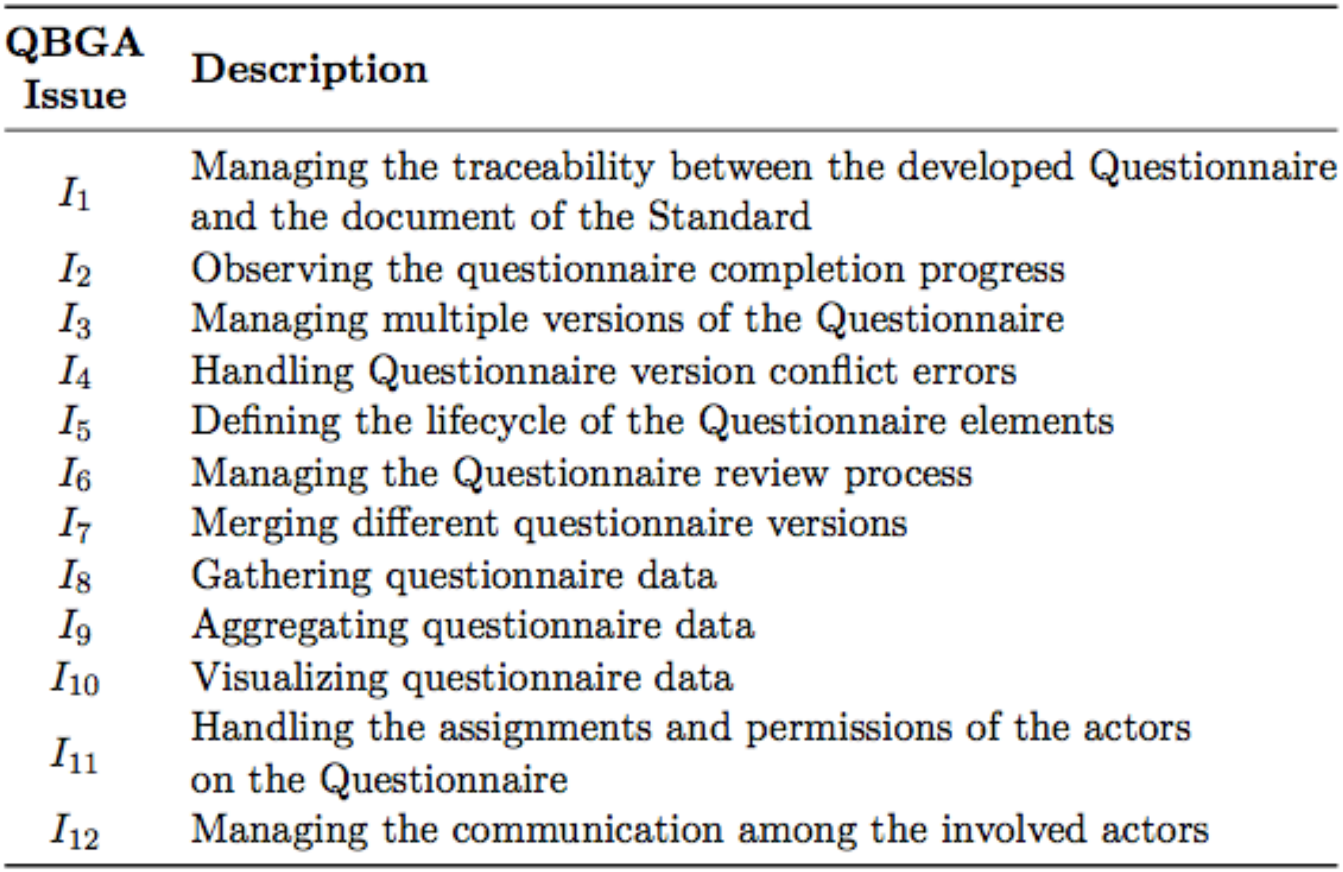
reverse.dieti.unina.it

@REvERSE_UNINA

vincenzo.desimone2@unina.it

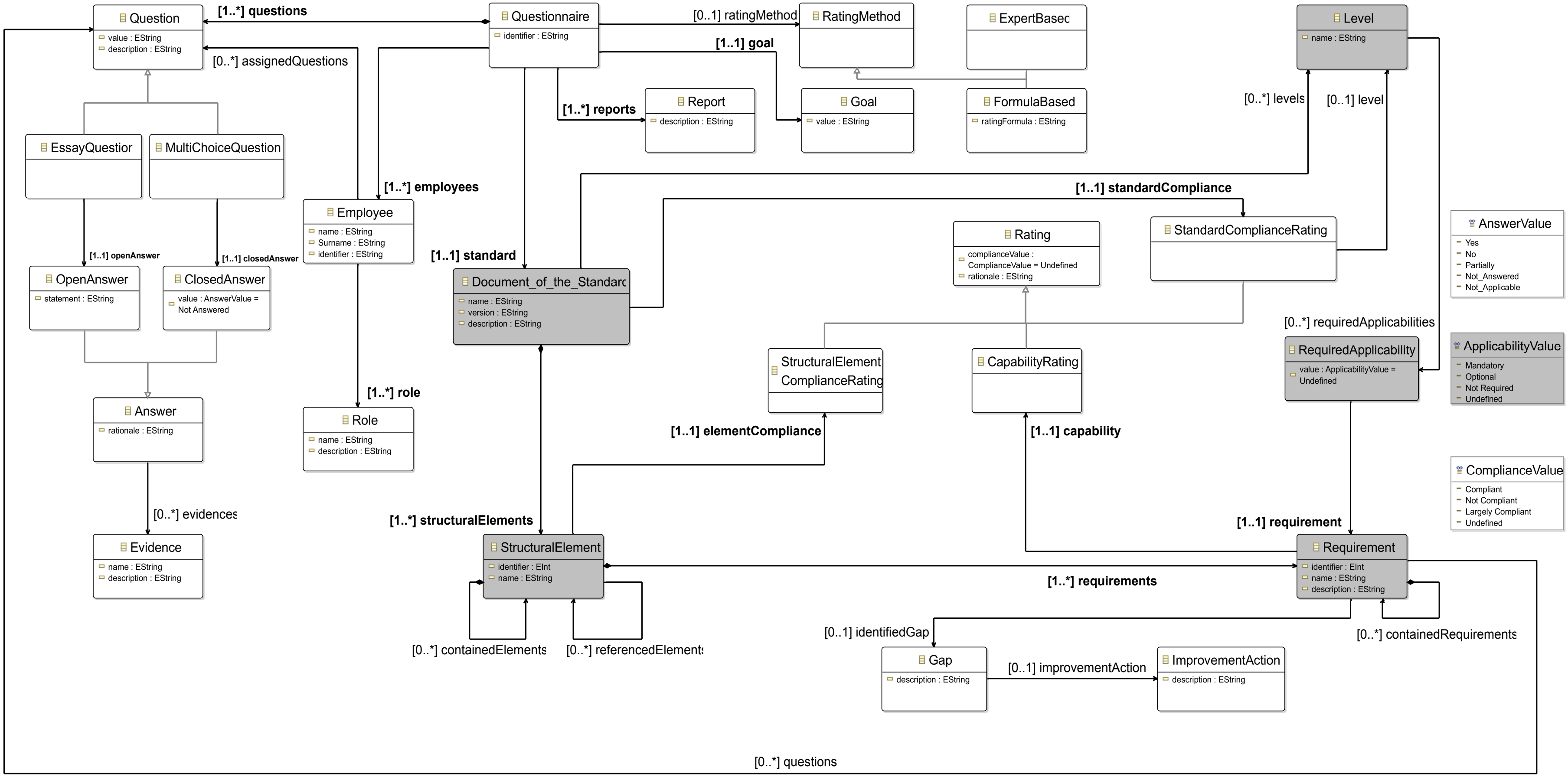
Backup Slides

Gap Analysis Process Issues

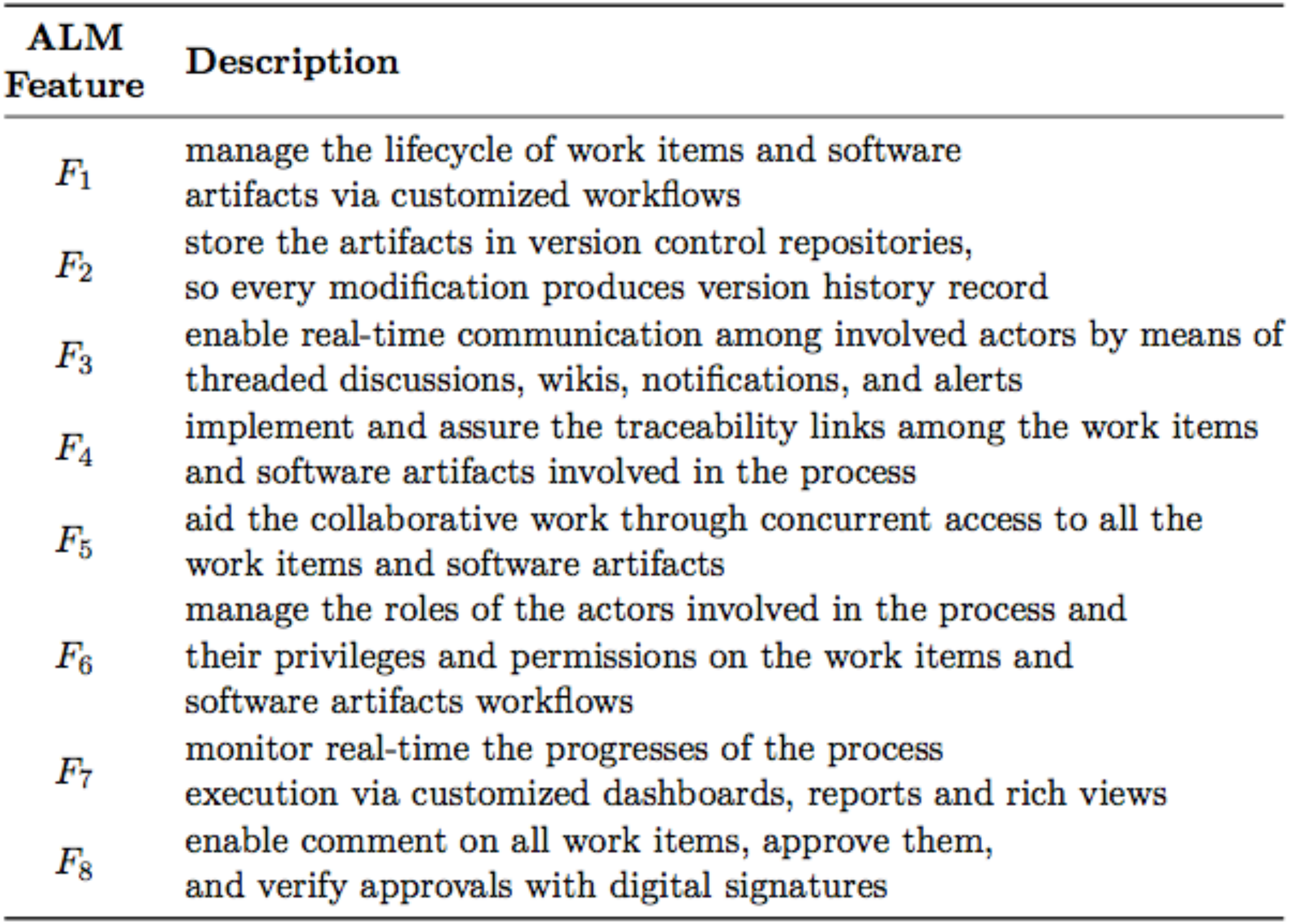


QBGA Issue	Description
I_1	Managing the traceability between the developed Questionnaire and the document of the Standard
I_2	Observing the questionnaire completion progress
I_3	Managing multiple versions of the Questionnaire
I_4	Handling Questionnaire version conflict errors
I_5	Defining the lifecycle of the Questionnaire elements
I_6	Managing the Questionnaire review process
I_7	Merging different questionnaire versions
I_8	Gathering questionnaire data
I_9	Aggregating questionnaire data
I_{10}	Visualizing questionnaire data
I_{11}	Handling the assignments and permissions of the actors on the Questionnaire
I_{12}	Managing the communication among the involved actors

Gap Analysis Processes Metamodel

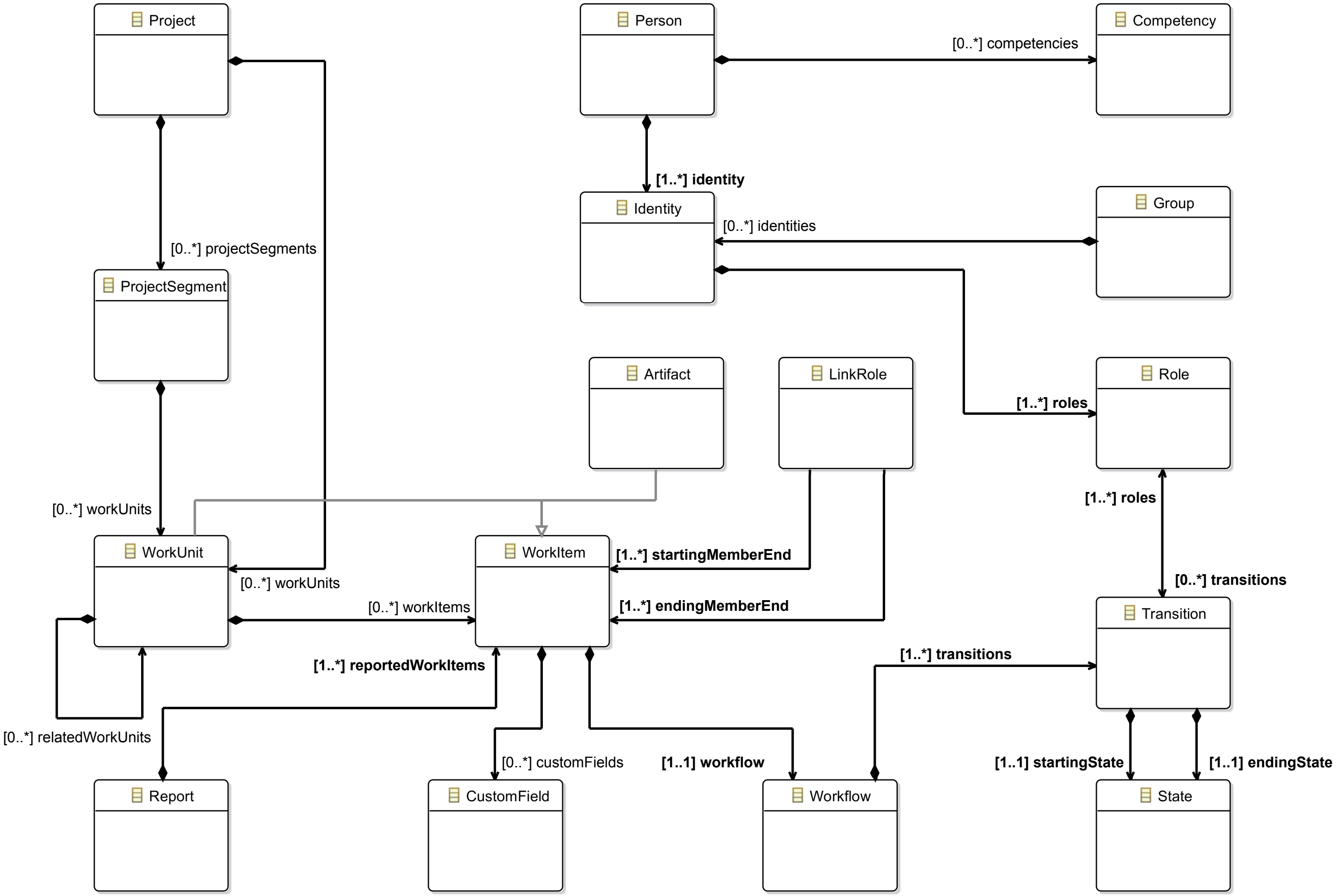


ALM Features

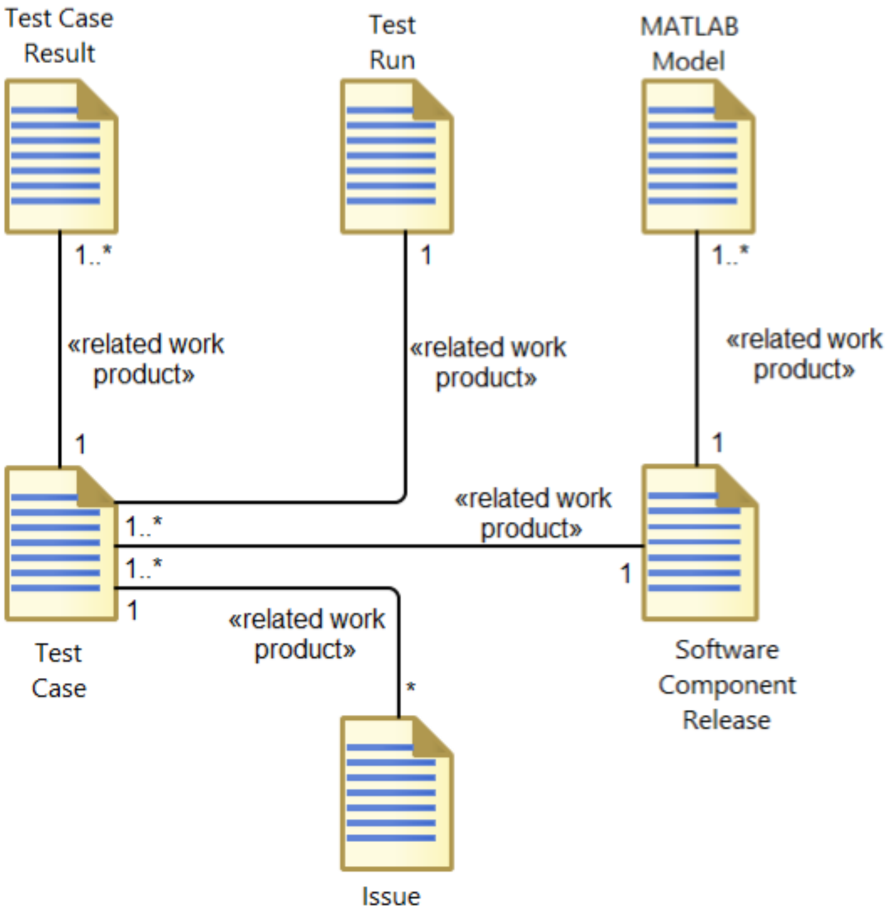


ALM Feature	Description
F_1	manage the lifecycle of work items and software artifacts via customized workflows
F_2	store the artifacts in version control repositories, so every modification produces version history record
F_3	enable real-time communication among involved actors by means of threaded discussions, wikis, notifications, and alerts
F_4	implement and assure the traceability links among the work items and software artifacts involved in the process
F_5	aid the collaborative work through concurrent access to all the work items and software artifacts
F_6	manage the roles of the actors involved in the process and their privileges and permissions on the work items and software artifacts workflows
F_7	monitor real-time the progresses of the process execution via customized dashboards, reports and rich views
F_8	enable comment on all work items, approve them, and verify approvals with digital signatures

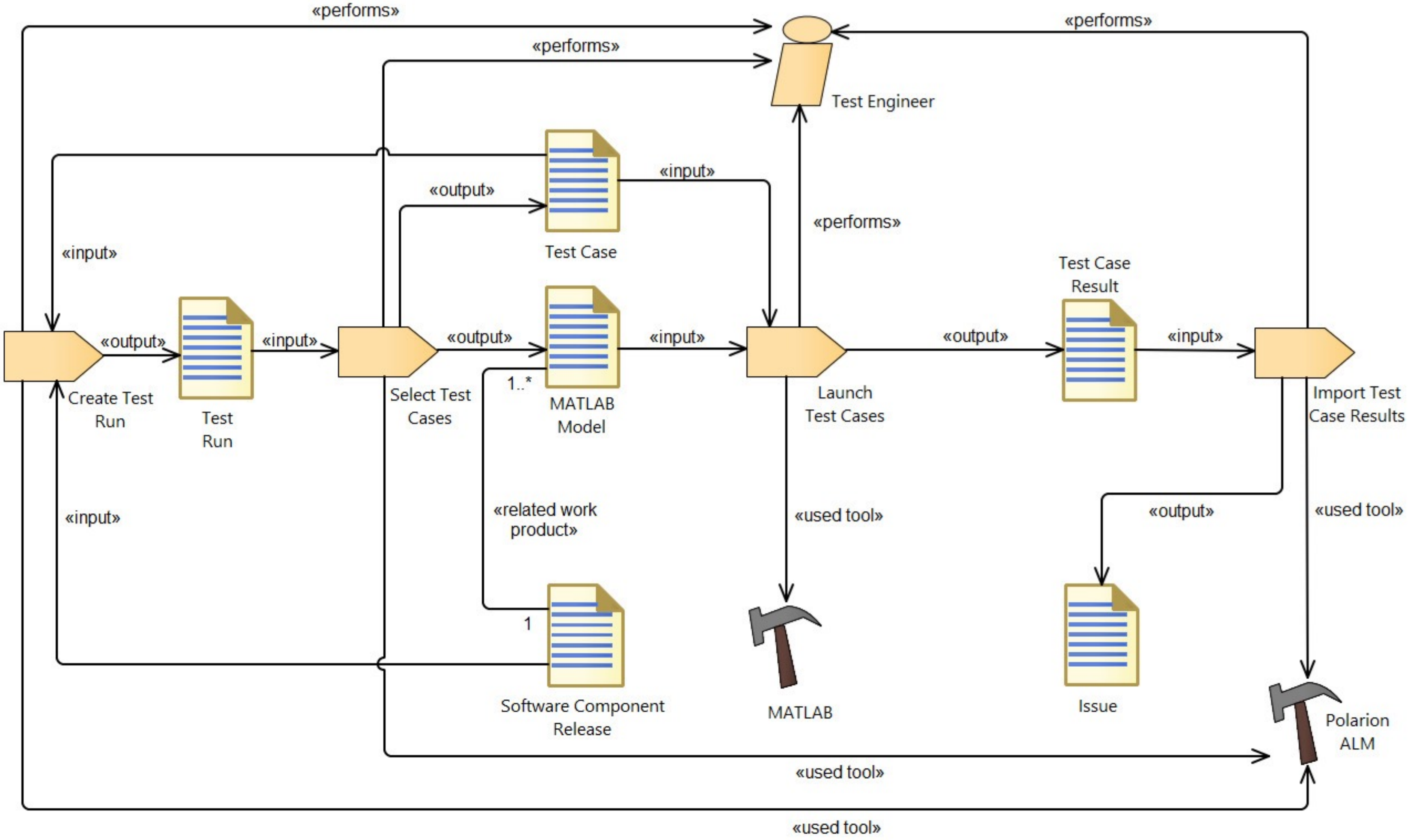
ALM Metamodel



Artifacts Model



Process Workflow


«performs»
«performs»
«performs»
Test Engineer
«input»
«output»
«performs»
Test Case
«input»
Test Case
Result
«output»
«input»
«output»
«input»
«output»
«input»
Create Test
Run
Test
Run
Select Test
Cases
1..*
MATLAB
Model
Launch
Test Cases
Import Test
Case Results
«input»
«related work
product»
«used tool»
«output»
«used tool»
1
Software Component
Release
MATLAB
Issue
Polarion
ALM
«used tool»
«used tool»