GRAN SASSO SCIENCE INSTITUTE
CENTER FOR ADVANCED STUDIES
Istituto Nazionale di Fisica Nucleare

MÄLARDALEN UNIVERSITY
SWEDEN

DATA EXTRACTION FORM

Execution of UML models

FEDERICO CICCOZZI (MÄLARDALEN UNIVERSITY)
IVANO MALAVOLTA (GRAN SASSO SCIENCE INSTITUTE)
BRAN SELIC (MALINA SOFTWARE CORP.)

DATA EXTRACTION FORM

Execution of UML models

February 26, 2016

ABSTRACT

This document describes the parameters of the data extraction activity of a systematic mapping study on the execution of UML models.

DOCUMENT VERSION CONTROL

Document status	**Version**	**Date**
Draft	0.1	August 18, 2015
Final	1.0	November 11, 2015

Contents

List of Figures

1 Characteristics conceptual model 10

List of Tables

1 Data extraction form 8
2 Application fields 9
3 Electronic databases and indexing systems considered in this research 9
4 Purposes of UML models execution 9
5 Research strategy (extracted from [1]) 11
6 Type of evidence categories 11
7 Type of evidence categories 12

Parameter name	Type	Multiple values	Category	Description
ID	*Int*		-	A study-specific unique identifier used across all the primary artifacts. It is also used as the name of the folder containing all the publications that were used in the data extraction in the primary study.
Assigned to	*String*		-	The name of the researcher performing the first extraction of the data from the primary study or tool.
Search method	$Set\{A, S, T\}$		**Search strategy**	The search method that the researcher uses for obtaining the study or the tool. A and S stand for $Automatic$ and $Snowballing$, respectively, whereas T is assigned when we are dealing with an industrial solution.
Automatic search source	*Set*, see Table 3 $\cup$ $\{other\}$		Search strategy	The name of the electronic database from which the study is obtained, $other$ if the study comes from an electronic database not included in Table 3.
Main study	*Int*		Search strategy	If the study comes from the snowballing activity, then this parameter represents the publication from which this primary study comes from (either backward or forward). If the study is an industrial solution, then this parameter is filled with the ID of the publication in which the tool is referenced.
Snowballing activity	$Set\{F, B\}$		Search strategy	If the publication comes from the snowballing activity, then this parameter represents in which kind of snowballing the publication has been included in our study. B and F stand for $backward$ and $forward$, respectively.
RQ1 - PUBLICATION TRENDS	Not applicable to industrial tools			
Title	*String*		**Publication trends (RQ1)**	Title of the primary study.
Authors	*String*	✓	Publication trends (RQ1)	List of the authors of the primary study.
Institutions	*String*	✓	Publication trends (RQ1)	List of the institutions of the primary study (as defined in the article itself).

Venue	*String*		Publication trends (RQ1)	The full name of the venue in which the study was published.
Venue acronym	*String*		Publication trends (RQ1)	The acronym of the venue in which the study was published (e.g., ICSE, ASE, FASE, MODELS, IROS, etc.).
Year	*Int*		Publication trends (RQ1)	The year of publication of the study.
Publication type	$Set\{C, J, B, W\}$		Publication trends (RQ1)	The type of publication in which the study was published. C = $Conference$, J = $Journal$, B = $Bookchapter$, W = $Workshop$,
Application domains	*Set*, see Table 2	✓	Publication trends (RQ1)	The application domain in which the proposed solution was applied (e.g., automotive, web applications, real-time embedded systems, etc.). The possible values of this parameter are derived from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
RQ2 - CHARACTERISTICS	Applicable to both research papers and industrial tools			
Approach				
Execution strategy	$Set\{T, C, I\}$		**Characteristics (RQ2)**	The type of execution strategy of the solution, according to the definitions provided in the introduction of the protocol of this study. T = $translation$, C = $compilation$, I = $interpretation$.
Simulation	Boolean		Characteristics (RQ2)	$True$ if the proposed solution allows its users to simulate UML models. Simulation is defined as executing a model in an environment that is different from the ultimate intended target environment. This can be done for a number of reasons, including the unavailability of the target environment, the availability of tools that are not present or not available in the target environment (e.g., debugging tools), or other pragmatic or economical reasons.
Intended benefits	Set, see Table 4	✓	Characteristics (RQ2)	The benefits that designers and developers using the proposed solution may gain when using it. The initial set of values comes from the preliminary keywording performed on the pilot studies.

Process	Boolean		Characteristics (RQ2)	$True$ if the proposed solution is closely associated with a specific development methodology.
Extensible	Boolean		Characteristics (RQ2)	$True$ if the proposed execution mechanisms can be extended or customized with additional components and capabilities, including those provided by third-party actors (e.g., plug-in based approaches, support for ecosystems of third-party modules, etc.).
Readiness level	$Set\{LOW, MEDIUM, HIGH\}$		Publication trends (RQ1)	The Technology Readiness Level (TRL), as defined by the systematic metric/measurement system for assessing the maturity of a particular technology [2], is defined as an integer n where $1 \leq n \leq 9$. In our data extraction activity we assign a LOW score if $n \leq 4$ (i.e., if the approach was either formulated, validated or demonstrated at most in lab), $MEDIUM$ if $5 \leq n \leq 6$ (i.e., if the approach was either validated or demonstrated in relevant environment), and $HIGH$ if $n \leq 7$ (i.e., if the approach was either completed, demonstrated, or proven in operational environment). It is important to note that the same measurement system is proposed by the Horizon 2020 European commission for the work program of the years 2014/2015[1].
Non-functional properties	$Set\{Performance, Security, Reliability, \ldots\}$	✓	Characteristics (RQ2)	The set of non-functional requirements supported by the proposed approach. In this case, by "supported" we mean that the proposed approach provides some mechanisms to assess, optimize, and/or reason about some specific non-functional requirement (e.g., the approach supports conducting some type of analysis of the energy consumed by the system, or its performance, security, etc.). The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
→ **Modeling**				

[1] http://ec.europa.eu/research/participants/data/ref/h2020/wp/2014_2015/annexes/h2020-wp1415-annex-g-trl_en.pdf

Partial models	Boolean		Characteristics (RQ2)	$True$ if the proposed solution supports the execution of incompletely specified or high-level (i.e., abstract) models (i.e., models in which there is a certain degree of uncertainty about whether specific parts of the system are part of the model itself [3]). This capability is particularly useful in early design phases where different approaches are proposed and evaluated. By supporting partial execution, it is possible to (1) significantly reduce the amount of time and effort required to evaluate an approach and (2) to increase the likelihood that unpromising approaches will be identified and discarded early in the design cycle.
UML diagrams	Set, see [4, Figure A.5]	✓	Characteristics (RQ2)	The subset of UML diagrams required by the proposed solution.
UML diagram types	$Set\{S, B\}$	✓	Characteristics (RQ2)	The types of UML diagrams used to produce executable models. We distinguish between *Structure* (S) and *Behaviour* (B) diagrams.
Action languages	$Set\{Alf, C++, Java, \ldots\}$	✓	Characteristics (RQ2)	The language used for specifying the fine-grained behaviors of the UML models [5]. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
Explicit UML profiles required	$Set\{MARTE, SysML, \ldots\} \cup ADHOC$	✓	Characteristics (RQ2)	The UML profiles which are needed by the proposed approach for executing the UML models. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.

Implicit UML profile	Boolean		Characteristics (RQ2)	$True$ if the proposed solution is based on an implicit profile, $false$ otherwise. Note that standard UML is not itself executable, due to the presence of numerous variability points and unspecified details (even fUML, a standardized executable subset of UML, has variability points). Hence, to execute UML models requires some additional semantic information to be included, representing a de facto UML profile. In many cases, these supplementary semantics are not clearly identified.
Modeling tool	$Set\{Papyrus, Rhapsody, \ldots\} \cup TOOL\_INDEP$		Characteristics (RQ2)	The modeling tool on which the proposed solution builds on (e.g., a plugin of the Papyrus modeling environment). The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
fUML	Boolean		Characteristics (RQ2)	$True$ if the proposed approach is based on the fUML standard by OMG[2], $false$ otherwise.
MDA levels covered	$Set\{PIM, PSM\}$	✓	Characteristics (RQ2)	By taking inspiration from the well-known MDA[3] modeling levels, this parameter focusses on the extent to which the proposed approach covers them. Additionally, as a result of the preliminary keywording we performed on the pilot studies, the set of possible values is extended andwith an explicit reference to whether the approach supports the modeling of the hardware for executing the model.
→ Execution				
Production system	Boolean		Characteristics (RQ2)	$True$ if models can be executed on the ultimate target platform (e.g., full code generation and execution), $false$ otherwise (e.g., models simulated in the modelling tool only).

[2] http://www.omg.org/spec/FUML/
[3] http://www.omg.org/mda/

Execution tools and technologies	$Set\{Acceleo, MDWorkbench, ATL, QVT, ad-hoc, \ldots\}$	✓	Characteristics (RQ2)	The names of all the tools and technologies used in order to make the UML models executable. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
Model-level debugging	Boolean		Characteristics (RQ2)	$True$ if the proposed solution provides some level of support for debugging at the level of UML model (e.g., support for specifying breakpoints in the models, step-by-step execution at the UML level, etc.), $false$ otherwise.
Formal specification languages	$Set\{CSP, graphs, Z, B, Petrinet, \ldots\}$	✓	Characteristics (RQ2)	The names of all the formal specification languages [6, §9.4.2] used in conjunction with the executable models (e.g., to generate test cases, or to formally verify an execution), if any. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
$\rightarrow\rightarrow$ **Translation**				
Intermediate artifacts	$Set\{configuration\ file, , DSL\ model, UML\ model, formal\ specification, \ldots\}$	✓	Characteristics (RQ2)	The type of intermediate artifacts that are produced between the source UML model and its generated executable. Intermediate artifacts may be fully automatically produced (e.g., autogenerated third-generation language source code), or manually refined by the designer. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.
Software platform	Boolean		Characteristics (RQ2)	$True$ when the generated code is meant to run on a specific software platform (e.g., OS, user-defined middleware, runtime libraries).
Translation targets	$Set\{Java, VHDL, C, \ldots\}$	✓	Characteristics (RQ2)	The target programming languages generated by the translation and later executed. The possible values of this parameter emerge from the keywording activity. The initial set of values comes from the preliminary keywording performed on the pilot studies.

Translation steps	Integer		Characteristics (RQ2)	The number of translation steps involved in the translation of the UML models. If the number of steps is variable, then we consider the maximum number of steps.
Traceability links support	Boolean		Characteristics (RQ2)	$True$ if the proposed approach includes explicit traceability links connecting the UML models and the outputs of their translation, $false$ otherwise. For example, such links are useful for debugging in cases where errors are encountered during execution or during compilation of the generated code.
→→ **Interpretation**				
Interpretation engine	$Set\{\ldots\} \cup \{ADHOC\}$		Characteristics (RQ2)	The name of the interpretation engine used for executing the UML models; its value can be $ADHOC$ if the runtime engine has been developed specifically for the proposed approach. The possible values of this parameter emerge from the keywording activity.
→→ **Compilation**				
Targets platform	String	✓	Characteristics (RQ2)	The names of the platform targeted by the binary code produced during the compilation step. The possible values of this parameter emerge from the keywording activity.
Compiler	$Set\{\ldots\} \cup \{ADHOC\}$		Characteristics (RQ2)	The name of the compiler (re-)used for compiling the UML model (e.g., GCC), $ADHOC$ if the compiler has been specifically developed for the proposed approach.
RQ3 - EVIDENCE	Not applicable to industrial tools			
Applied research methods	Set, see [7, fig. 19]	✓	**Evidence (RQ3)**	The type of applied research method for giving evidence about the approach. Based on [7, fig. 19], possible values of this parameter are *Validation* ($VALID$) and *Evaluation* ($EVAL$).
Type of evidence	Paper fragment		Evidence (RQ3)	Set of fragments of the study in which the authors describe the type of evidence of the proposed solution.
Type of evidence by category	Set, see Table 7		Evidence (RQ3)	Assigned code for the type of evidence of the proposed solution (see the codes in Table 7).

Evaluation systems	Paper fragment	✓	Evidence (RQ3)	Set of fragments of the study in which the authors describe the concrete systems they model and execute for validating the proposed solution.
Types of evaluation systems	Open set	✓	Evidence (RQ3)	Type of concrete system used in the study for validating the proposed solution.
RQ4 - LIMITATIONS	Not applicable to industrial tools			
Limitations and unsolved challenges	Paper fragment		**Limitations (RQ4)**	Set of fragments of the study in which the authors discuss the limitations and unsolved challenges of their approach.
Type of limitations and unsolved challenges	Open set		**Limitations (RQ4)**	Codes extracted during the data analysis phase on the limitations and unsolved challenges of the approach.
NOTES				
Notes	*String*		-	Free field in which the data extractor keeps track of potentially relevant information about the research study or industrial tool.

Table 1: Data extraction form

ID	Description
GENERIC	The primary study is not tailored to any specific application field
SPECIFIC	The primary study is tailored to a specific application field

Table 2: Application fields

Name	Type	URL
IEEE Xplore Digital Library	Electronic database	http://ieeexplore.ieee.org
ACM Digital Library	Electronic database	http://dl.acm.org
SCOPUS	Indexing system	http://www.scopus.com
Web of Science	Indexing system	http://webofknowledge.com

Table 3: Electronic databases and indexing systems considered in this research

ID	Description
PROD	The overall software development process is faster with respect to the manual development of the system.
QUALITY	The system being executed is ensured to satisfy some specific quality attributes, such as performance, security, maintainability, etc.
CORRECT	The focus of the UML model execution is on checking the functional correctness of the system.

Table 4: Purposes of UML models execution

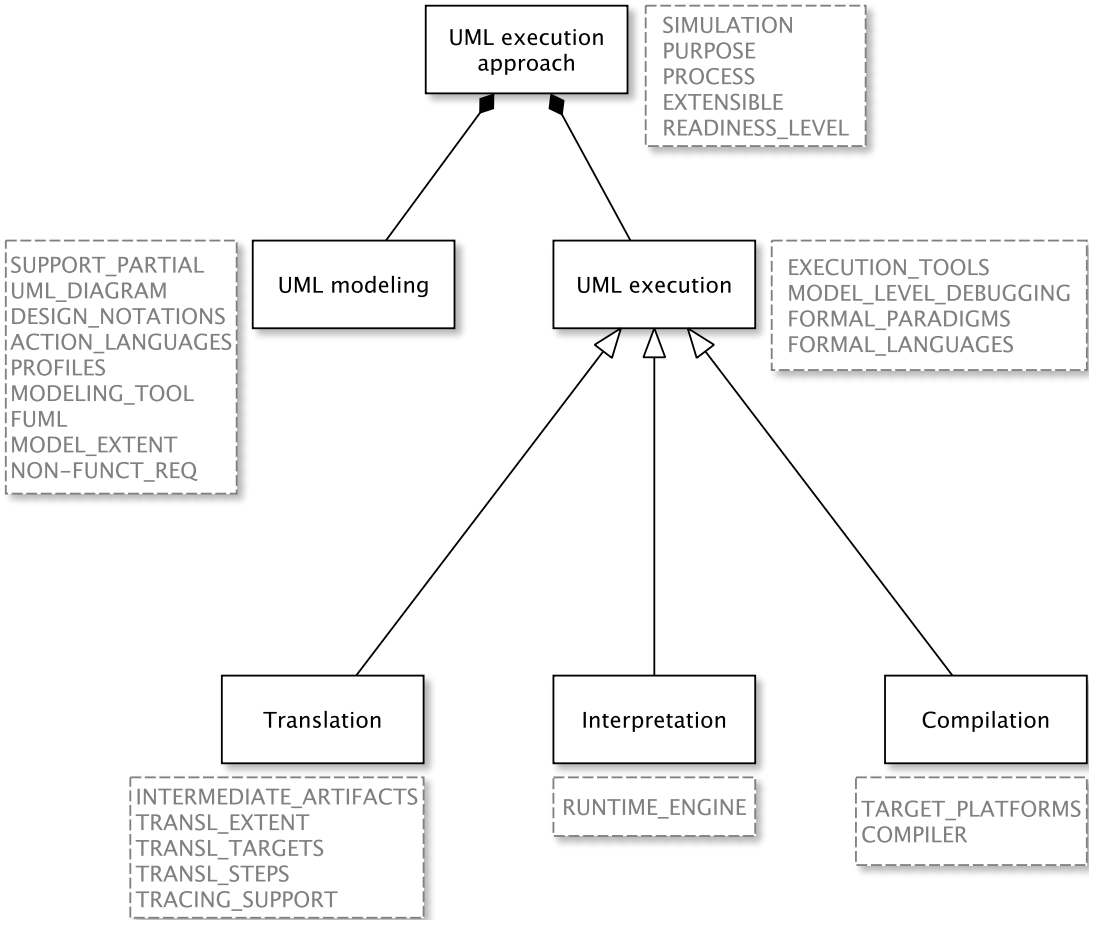

Figure 1: Characteristics conceptual model

Category	Description
Validation Research	Techniques investigated are novel and have not yet been implemented in practice. Techniques used are for example experiments, i.e., work done in lab.
Evaluation Research	Techniques are implemented in practice and an evaluation of the technique is conducted. It shows how the technique is implemented in practice (solution implementation) and the consequences of the implementation in terms of benefits and drawbacks (implementation evaluation).
Solution Proposal	A solution for a problem is proposed, the solution can be either novel or a significant extension of an existing one. The potential benefits and the applicability of the solution is shown through a small example or a good line of argumentation.
Philosophical Papers	These papers sketch a new way of looking at existing things by structuring the field in form of a taxonomy or conceptual framework.
Opinion Papers	These papers express the personal opinion of somebody whether a certain technique is good or bad, or how things should been done. They do not rely on related work nor research methodologies.
Experience Papers	They explain on what and how something has been done in practice. It has to be the personal experience of the author.

Table 5: Research strategy (extracted from [1])

Category	Description
EXAMPLE	1 in-house example (includes real-life examples).
SET_OF_EXAMPLES	several in-house (not industrial) examples (includes real-life examples), several runs and comparisons of one example, even several models for the system, e.g. with different sizes.
EMPIRICAL_LAB	case-studies, controlled experiments and empirical evaluations in lab.
INDUSTRIAL_ EXAMPLE	example coming from industry and performed in lab.
SET_OF_INDUSTRIAL_ EXAMPLES	several industrial examples (even several models for the system, e.g. with different sizes).
EMPIRICAL_ INDUSTRIAL	case-studies, controlled experiments and empirical evaluations either in industry or in real-world scenarios.
INDUSTRIAL_ EVALUATION	evaluation performed by industrial actors, solution used in industry.

Table 6: Type of evidence categories

References

[1] K. Petersen, R. Feldt, S. Mujtaba, M. Mattsson, Systematic mapping studies in software engineering, in: Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering, EASE'08, British Computer Society, Swinton, UK, UK, 2008, pp. 68–77. URL http://dl.acm.org/citation.cfm?id=2227115.2227123

[2] J. C. Mankins, Technology readiness levels, White Paper, April 6.

[3] M. Famelis, R. Salay, M. Chechik, Partial models: Towards modeling and reasoning with uncertainty, in: Software Engineering (ICSE), 2012 34th International Conference on, IEEE, 2012, pp. 573–583.

[4] UML Superstructure, version 2.4.1 .

[5] S. J. Mellor, S. Tockey, R. Arthaud, P. Leblanc, An action language for uml: proposal for a precise execution semantics, in: The Unified Modeling Language. UML'98: Beyond the Notation, Springer, 1999, pp. 307–318.

[6] P. Bourque, R. E. Fairley, et al., Guide to the Software Engineering Body of Knowledge (SWEBOK (R)): Version 3.0, IEEE Computer Society Press, 2014.

[7] K. Petersen, S. Vakkalanka, L. Kuzniarz, Guidelines for conducting systematic mapping studies in software engineering: An update, Information and Software Technology 64 (2015) 1–18.