Quality in Model-Driven Engineering

A tertiary study

Miguel Goulão · Vasco Amaral · Marjan Mernik

Abstract *Context:* Model-Driven Engineering (MDE) is believed to have a significant impact in software quality. However, researchers and practitioners may have a hard time locating consolidated evidence on this impact, as the available information is scattered in several different publications.
Objective: Our goal is to aggregate consolidated findings on quality in MDE, facilitating the work of researchers and practitioners in learning about the coverage and main findings of existing work as well as identifying relatively unexplored niches of research that need further attention.
Method: We performed a tertiary study on quality in MDE, in order to gain a better understanding of its most prominent findings and existing challenges, as reported in the literature.
Results: We identified 22 systematic literature reviews and mapping studies and the most relevant quality attributes addressed by each of those studies, in the context of MDE. Maintainability is clearly the most often studied and reported quality attribute impacted by MDE. 80 out of 83 research questions in the selected secondary studies have a structure that is more often associated with mapping

Miguel Goulão
NOVA LINCS, Departamento de Informática
Universidade Nova de Lisboa
Tel.: +351-21-2948536
Fax: +351-21-2948541
E-mail: mgoul@fct.unl.pt

Vasco Amaral
NOVA LINCS, Departamento de Informática
Universidade Nova de Lisboa
Tel.: +351 212948536
Fax: +351 212948541
E-mail: vma@fct.unl.pt

Marjan Mernik
Faculty of Electrical Engineering and Computer Science
University of Maribor
Tel.: +386 22207455
Fax: +386 22207272
E-mail: marjan.mernik@um.si

existing research than with answering more concrete research questions (e.g. comparing two alternative MDE approaches with respect to their impact on a specific quality attribute). We briefly outline the main contributions of each of the selected literature reviews.

Conclusions: In the collected studies, we observed a broad coverage of software product quality, although frequently accompanied by notes on how much more empirical research is needed to further validate existing claims. Relatively little attention seems to be devoted to the impact of MDE on the Quality in Use of products developed using MDE.

Keywords Quality · Model-Driven Engineering · Tertiary Study

1 Introduction

Model-**D**riven **E**ngineering (MDE) (Schmidt 2006; da Silva 2015) is an increasingly popular software development methodology that makes use of rigorous abstractions, known as models (that allow predictions or inferences to be made), to design and implement a given system. Models are used to fight the complexity of specifying systems for nowadays target platforms, not tackled by the general purpose programming languages, and express domain concepts effectively. Therefore, together with model transformation approaches (i.e. model-to-model and model-to-code), that promote automatic or semiautomatic generation of code, models are no longer just means of documentation but they are a core development artifact of the system. Models are commonly specified in visual modeling languages, e.g. **U**nified **M**odeling **L**anguage (UML) (OMG 2015), but can also be specified in dedicate textual languages, like Domain Specific Languages, as long as they are able to abstract concepts from the domain of the problem instead of the domain of the solution (e.g. RubyTL (Cuadrado et al 2006)).

MDE is often claimed to bring several benefits to software projects that, ultimately, allow dealing with the increasing development complexity of the software systems, leading to cost reductions in the development and evolution processes, while also increasing the quality of the target developed software. The apparent relatively low level of adoption of MDE in industry does not seem to be in line with these benefits. A growing number of studies (mostly surveys with practitioners) is shedding light to the actual usage of MDE, as well as some of the challenges MDE is still facing, hampering a wider adoption by practitioners. In this paper we are interested to observe the status of both perspectives regarding quality of the product and process.

Some reports point to a relatively low adoption level of modeling (defined as the activity of designing system models in a given language). (Forward et al 2010) notes that models are mostly created for documentation and up-front design, but not so much for code generation partly due to the predominance of code-centric environments, rather than modeling-centric ones. More recently, (Hutchinson et al 2014) reported a stronger usage of code generation. Nevertheless, the prevalence of code-centric projects is visible, for example, in open-source projects (Badreddin et al 2013). Although the UML is often claimed to be a *de facto* standard in industry, a recent survey reported that only 15 out of 50 practitioners use it (Petre 2013).

Another survey (Fernández-Sáez et al 2015) focused on how UML documentation is used to support maintenance tasks in industry and found a wider adoption level (43% of the respondents use it, and an additional 16% use an alternative language for that purpose, with a set of 178 practitioners from 12 different countries). In some countries, like Italy, MDE is becoming much more popular, with 13% of companies working with modeling always, and an additional 53% doing it sometimes, in a survey conducted with 155 practitioners (Torchiano et al 2013).

There may be several reasons contributing to these challenges when it comes to MDE adoption. Thanks to the technological advances and scientific investment in MDE in the last decade, we are now reaching a level of maturity that allows us to observe the current status and understand its impact in systems development and understand what can be hampering a wider adoption. (Whittle et al 2015) reports a summary of interviews with the practitioners' perceptions about the challenges of the current state of practice with MDE tools. Here, several quality factors are recurrently mentioned, namely: the quality of the generated code; the excessive complexity of tools and the abstraction mismatch between the tools offered (including languages) and the users; the impact of the tools in the quality of the product and processes, where productivity and maintainability are the major concerns, e.g. how well tools integrate in the process. (Agner et al 2013) reports a survey on 209 Brazilian software engineers, concluding there is a strong perception of the relevance of MDE in the promotion of quality, although the level of adoption is much lower. The result of this study puts into evidence the quality as a key challenge of MDE. In this paper we will select and consolidate existing work on the impact of MDE in quality.

Despite the relevance of this perspective we should not ignore other adoption challenges of MDE. For example, (Ameller et al 2015) reports on an ongoing international study to assess the way MDE is being used in industry to cope with non-functional requirements, so that the main shortcomings of existing approaches, as perceived by practitioners, can be transformed into significant improvement opportunities.

Others, mention social or organizational challenges (Hutchinson et al 2011, 2014) like resistance to change (Mohagheghi et al 2013). (Cuadrado et al 2014) presents lessons learned while introducing MDE in small companies, and notes that most existing studies are performed in medium and large organizations. This is problematic due to the high number of smaller companies which could benefit from adopting MDE, but may find it hard to uncover compelling information supporting that decision.

The challenges of MDE adoption may even have a pedagogic/training nature. Industry representatives have reported the difficulty of hiring well-trained MDE practitioners (Whittle et al 2015). (Badreddin et al 2015) reports on a survey conducted in three universities from the US, Canada and Israel, where students' perceptions on the value of modeling decrease as students progress in their degree training. Additionally, (Fernández-Sáez et al 2015) mentions the lower performance with geographically distributed collaborative work.

All these studies aim, to some extent, to mitigate a perceived problem of lack of consolidated evidence that is compelling enough so that industry can fully understand the strengths, but also the challenges and shortcomings, of MDE adoption. These evidences can be collected from surveys as those reported earlier, or, more generally, from other primary studies on MDE. In this paper, we are particularly

interested in the evidence concerning the impact of MDE adoption in the quality of the products built with it (and in the quality of the process of building and evolving those products with MDE). The area is too broad for a single study, but there are already several secondary studies, in the form of literature reviews that partially address it. In this paper we leverage those secondary studies, by conducting a tertiary study to provide an overview of how MDE impacts quality.

The main contributions of this paper are:

– **C1:** A mapping of the most representative secondary studies that cover quality in MDE, their origin (to identify key players in this research area), and the quality attributes addressed by each of the secondary studies. This mapping is expected to serve as a starting point for researchers and practitioners interested in quality in MDE to locate relevant consolidated reviews on the area, and the experts responsible for those reviews.
– **C2:** An annotated overview of the existing aggregated information on quality in MDE. By outlining the main contributions of the included secondary studies, we provide a single resource from which researchers and practitioners can start exploring further work.
– **C3:** A report on the level of consolidation of the aggregated information on quality in MDE. One of the roles that a literature review can play is to aggregate and consolidate information that would be otherwise scattered in several different publications, providing a better overview of the area, while preserving traceability links to the original research sources. In this tertiary review, we are interested in the extent to which this has been achieved in the reported literature reviews.

This paper is organized as follows. Section 2 reports the methodology followed to conduct our literature review. Section 3 reports data collected from each of the selected secondary studies. Section 4 answers our research questions, and reports the limitations of our study. Finally, section 5 reports our conclusions from this tertiary study.

2 Research Method

This paper can be characterized as a work in **E**vidence-**B**ased **S**oftware **E**ngineering (EBSE) (Kitchenham et al 2004). EBSE aims at collecting the best available evidence to address software engineering research questions, both from practitioners and researchers. Typically, this is performed by aggregating existing empirical studies (referred to as primary studies) on a particular topic and performing a literature review on them, in an unbiased way. This is often done through a **S**ystematic **L**iterature **R**eview (SLR). With enough primary studies available, one should aggregate the evidence collected from them to answer the specified research questions (e.g. Is technique A more effective than technique B under a specific context?), by performing a statistical meta-analysis on those evidences. Unfortunately, it is relatively rare to find good meta-analysis opportunities in software engineering, due to the lack of sufficient primary studies on the phenomenon under scrutiny. Often, we will find more coarse-grained research questions aimed at mapping the existing knowledge on a specific topic rather than aggregating it through meta-analysis. These questions are addressed in **S**ystematic **M**apping **S**tudies (SMSs),

which are also a systematic form of literature review. They differ from conventional SLRs in that they are aimed at classifying and performing a thematic analysis of their topic, with more generic research questions, a broader scope, a more inclusive search process, and resulting in a categorization of existing work in several dimensions (see, for example, (Kitchenham et al 2011) for a pragmatic comparison between SLRs and SMSs). SMSs can be performed prior to SLRs, to help identifying research questions which are good candidates for data aggregation. Both SLRs and SMSs are considered secondary studies, as they review primary studies.

This work can be classified as a *tertiary study*. It was conducted following the guidelines on SLRs (and SMSs), detailed in (Kitchenham and Charters 2007). We can regard this work as a systematic mapping study targeted at secondary studies (*i.e.* SLRs and SMSs), rather than at primary studies, in order to gather consolidated evidence from those secondary studies, hence being a tertiary study. Our goal, with this decision, was to gather a broader overview on the current state of the art concerning the research and practice of quality in MDE.

2.1 Research questions

The research questions addressed in this study are:

- **RQ1: What is the currently available information concerning quality in MDE, systematically aggregated by means of a SLR, or a SMS?** Our goal is to characterize the current state of the art in this research domain, providing readers with pointers for looking up additional information concerning each particular quality characteristic and how it relates to MDE.
  - *RQ1.1: How many primary studies are included in these reviews?*
  - *RQ1.2: What is the time span covered by these reviews?*
  - *RQ1.3: Is the list of the included primary studies available?*
  - *RQ1.4: Is the quality of the primary studies assessed? If so, how is this done?* This assessment is often *ad-hoc*, but can also follow guidelines, such as those in (Kitchenham and Charters 2007).
  - *RQ1.5: Who is the secondary study targeted to?* The target audience can be made of *researchers*, *practitioners*, or *both*.
  - *RQ1.6: What kind of software engineering task is the secondary study aimed at?* These include *decision support*, *knowledge support* and *scoping*.
  - *RQ1.7: What is the object of study in the included secondary studies?* Typically, this includes *models*, *transformations*, or the *software process* (at least one of these, although for several papers, more than one of these objects of study are addressed).
  - *RQ1.8: Which quality attributes are addressed in the secondary study?* These include product quality and quality in use attributes, as defined by the ISO/IEC 25010:2011 standards (ISO/IEC 2011).
- **RQ2: What is the current status of consolidation of data collected from different literature reviews covering quality in MDE and made available through published secondary studies?** Ideally, one of the potentially key benefits to be taken from literature reviews (in particular, from SLRs) is to be able to aggregate data collected independently in different studies.

- **RQ3: Who are the key players in consolidating knowledge on quality in MDE through literature surveys?** This question is of a demographic nature. Our goal is to characterize the community in this research domain. To answer this question, we break it down into more refined ones, as follows:
  - *RQ3.1: How many systematic reviews, including systematic literature reviews and mapping studies, addressing the topic of quality in MDE, are available?* This provides a quick overview on the liveliness of the community which is consolidating information on this research topic.
  - *RQ3.2: What is the origin of these reviews?* This covers organizations and countries, and is aimed at better understanding the extent to which quality in MDE is becoming a global concern.
  - *RQ3.3: What is the impact of these reviews, in terms of citations?* This will provide some insight on the "popularity" of this research topic.

2.2 Search process

We searched two digital libraries, namely IEEExplore and ACM Digital Library, and the SCOPUS indexing system. All searches were based on title, keywords and abstract. In IEEExplore and ACM digital libraries, full text was also searched.

The canonical search string, which was adapted for coping with the different syntax rules required by each search engine was as follows:

```
("model driven" OR "MDE" OR "MDD") AND
("systematic review" OR "literature review" OR
 "literature survey" OR "survey" OR
 "overview of research" OR "mapping study" OR "review")
```

Our secondary studies selection process started by running the search string in the three repositories. We collected the search results, and merged them into a single spreadsheet, keeping track of the origin of each of the papers. The results yielded 481 candidate papers. These papers were then passed on to the next step of our selection process.

2.3 Inclusion and exclusion criteria

We defined objective inclusion and exclusion criteria, so that the selection process could be conducted in a consistent way. These inclusion and exclusion criteria were applied to the 481 candidate papers. The inclusion criteria are as follows:

- **IC1:** The paper is a peer-reviewed full paper.
- **IC2:** The paper includes a literature review where the papers were included using a defined search process – e.g. a defined search string on an explicit set of digital libraries, or a manual search conducted on the proceedings of a specific venue, within a defined time interval (for example, a manual search on the **I**nternational **C**onference on **S**oftware **E**ngineering (ICSE) conference series, from 2000 to 2014).
- **IC3:** The paper is related to model-driven development, or engineering, and explicitly covers quality in that context.

– **IC4:** The paper is written in English.

The exclusion criteria provide a complementary view on the inclusion criteria and are as follows:

– **EC1:** The paper is not a peer-reviewed full paper. This excludes presentations, extended abstracts, white papers, introductory forewords for conferences or journal special issues, non peer-reviewed book chapters, and books, to name some of the most common examples.
– **EC2:** The paper reviews primary studies following an undefined studies selection process.
– **EC3:** The paper is out of scope.
– **EC4:** The paper is not written in English.
– **EC5:** The paper is subsumed by another paper which is already included in our literature review.
– **EC6:** The paper is a duplicate of another paper in the sample.
– **EC7:** The paper does not include a secondary study.

The first exclusion criteria used after the automated search was **(EC6)**. As expected, some papers were indexed by more than one repository and, therefore, duplicates. We excluded all duplicates, keeping track of which repositories contained which papers, originally.

Table 1 summarizes the key statistics on our automatic search, using our search string, followed by the application of **(EC6)**. We present the individual totals for each of the repositories searched, their sum, the number of removed duplicates, and the total number of candidate papers, after removing duplicates.

Table 1 Candidate papers found through automated search

ACM	IEEExplore	SCOPUS	Sum	Removed duplicates	Total
127	96	258	481	-128	353

We then proceeded to screening the papers titles and abstracts, to detect papers that should remain as candidates for further analysis. Each paper was screened by at least one author. We were very conservative in this first assessment. Whenever we suspected the paper might contain a (possibly systematic) literature review, or mapping study, we kept it for further analysis. This conservative approach was chosen to mitigate the risk of missing important studies. Indeed, it is relatively common for software engineering abstracts to have completeness and clarity shortcomings, as discussed in (Budgen et al 2008), which make this screening more error prone, when compared to what can be achieved with structured abstracts, as the one used in this paper. In cases where we were inclined to reject a paper, but not totally confident about this decision, we marked it for a second review by a different author. We only rejected directly papers during title and abstract screening when we were very confident they were not secondary studies, using **(EC1)**, or when they were obviously out of scope, using **(EC3)**. This process eliminated a total of 310 papers out of the 353 candidate papers. Of the excluded 310 papers, 42 were not peer review papers (e.g. our search string caught several foreword chapters in conference proceedings), while the remaining papers were out of scope. Out

of scope papers include papers on unrelated topics that somehow made it through the automated search and, more commonly, papers that are within our topic, but do not include secondary studies **(EC7)**, thus being inadequate for inclusion as part of the list of papers scrutinized in a tertiary study.

This process lead to selecting 43 papers for full text review and, if eligible, for data extraction. All these papers were reviewed by at least one of the authors. 8 of these papers were then excluded for not being a systematic review. Typically, these were informal literature surveys, from where it was impossible to recover the primary study selection process, as expressed by **(EC2)**. 14 papers were excluded for being out of scope, following **(EC3)**. These papers would typically have a systematic literature review, or mapping study, but their research questions were not aligned with our own. Finally, 2 papers were excluded because they were preliminary secondary studies later extended in other papers, following **(EC5)**. We kept the most complete version of these papers for further scrutiny.

In the end of the full text data extraction, we had 19 papers, plus 4 identified through snowball, by checking the references in the selected papers. 1 of the papers identified through snowball was subsumed by a second paper, also identified in the snowball process, so it was discarded. We ended with 22 secondary studies that we will analyze in the remaining of this paper.

Table 2 summarizes the included and excluded papers in each phase and the reasons for paper exclusion from our final data set.

Table 2 Included and excluded papers in each phase

Phase	Included	Excluded	Remaining
Automated search	**481**		
Duplicates removal (EC6)		**128**	**353**
Title + Abstract screening		**310**	**43**
- Not a peer reviewed paper (EC1)		(42)	
- Out of scope (EC3)		(268)	
Full-text reading / Data extraction		**24**	**19**
- Not a systematic review (EC2)		(8)	
- Out of scope (EC3)		(14)	
- Subsumed by another paper (EC5)		(2)	
Snowball search	**4**		**23**
Full-text reading / Data extraction		**1**	**22**
- Subsumed by another paper (EC5)		(1)	
Final set of included secondary studies			**22**

Concerning the completeness of this paper identification process, we started with informal searches, along with a set of papers we were aware of. After this process, all the previously identified papers had been included, which gives us some confidence on the completeness of this work. Moreover, as we used an additional snowball search step, built on the references lists in the included papers, we expect most of the relevant papers were detected in this aggregated search. All exclusions during the full text reading were double checked by a different author, in order to minimize the risk of missing a relevant paper.

2.4 Quality assessment

The confidence we can place in the results of a systematic review is largely dependent upon the quality of the evidence collected in that review (Dybå and Dingsøyr 2008; Zhou et al 2015). The same holds for a tertiary study, such as this one, where appraising the quality of the secondary studies selected for our review is extremely relevant.

The so-called DARE criteria[1] are often used in the context of software engineering – see, for examples, a relaxed version of DARE in (Kitchenham et al 2010) and a complete version in (Cruzes and Dybå 2011)). In this paper we use both versions in the appraisal of the quality of the selected papers, keeping the evaluation criteria and scales taken from (Cruzes and Dybå 2011; Kitchenham et al 2010), for easier comparison with other tertiary studies adopting these criteria. In the original DARE criteria, used in (Cruzes and Dybå 2011), reviews have to meet at least four of these criteria, and the first three are mandatory. In the more relaxed version of the DARE criteria, used in (Kitchenham et al 2010), only 3 out of 5 criteria are required, with only criteria 1 and 2 as mandatory. We describe those criteria in detail in section 3.2, where we then apply them to analyse the quality of the included secondary studies.

2.5 Data extraction process

The data extraction process was conducted by the three authors. After selecting the candidate papers, these were split among us, so that one of us would read the paper, extract information from it, and store the collected data in a spreadsheet shared in the cloud by the three authors. Whenever some detail of the data extraction was unclear, we marked it to make sure a second author would double check it. We also kept comments on the shared spreadsheet so that the rationale for decisions and classification of the data collected from the paper was recorded and discussed in our team meetings. In practice, all selected papers were read by at least two of the authors, the first for data extraction and the other(s) for double-checking and helping in situations where the secondary studies information was perceived as less clear.

2.6 Extracted data

In this section we describe the main information gathered from each secondary study:

- Basic identification information, including authors, their affiliations and countries, source in which the secondary review was published, year of publication and number of citations (taken from Google Scholar).
- The starting and ending years of publication of the primary studies covered in the secondary study.

---

[1] The **DARE** criteria are based on those used by the Center for Reviews and Dissemination of the University of York, for assessing the eligibility of systematic reviews to be included in their **D**atabase of **A**bstracts of **R**eviews of **E**ffects (DARE). `http://www.crd.york.ac.uk/CRDWeb/`

- The number of primary studies included in the secondary study.
- Information on the availability of the list of included primary studies.
- Information concerning if and how (i.e. according to which criteria) the quality of the included primary studies was assessed.
- The target audience (as reported by its' authors) for the secondary study, which may include researchers, practitioners, or both.
- The secondary study's intended use, as reported by its authors. In particular, we classify secondary studies as being aimed at decision support, knowledge support, or scoping.
- The objects of study considered in the secondary study, namely models, transformations, and the software process itself.
- The quality models used as reference in the secondary study, if any.
- The product quality model attributes addressed in the secondary study, if any.
- The quality in use model attributes addressed in the secondary study, if any.

The last two items this list (quality model and quality in use attributes addressed by the secondary study) were categorized using standard quality models from ISO/IEC. We use the standard ISO/IEC 25010:2011 quality model (ISO/IEC 2011), in particular its model *product quality* and its *quality in use* as references. In both cases, quality concerns both the quality of the products built with MDE and of the tools and methods supporting MDE itself.

The product quality model covers both static properties of a software and the dynamic properties of the computer system in which it runs. Using this model allows us to reason about the existing research concerning product quality attributes and how they are regarded in MDE research, according to the selected secondary studies. This quality model, presented in figure 1, includes 8 attributes, which are then further decomposed into sub-attributes. Here we describe the higher level attributes, which we will use to guide our discussion on the coverage of quality attributes in the included secondary studies.

- *Functional suitability* is the capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions.
- *Reliability* is the capability of the software product to maintain a specified level of performance when used under specified conditions.
- *Performance efficiency* is the capability of the software product to provide appropriate performance, relative to the amount of resources used, under stated conditions.
- *Usability* is the capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions.
- *Security* is the capability of the software product to protect information and data and controls the access level of people, products, or systems.
- *Compatibility* is the capability of a product, system, or component, to exchange information with other products, systems and components.
- *Maintainability* is the capability of the software product to be modified. Modifications may include corrections, improvements or adaptation of the software to changes in environment, and in requirements and functional specifications.
- *Portability* is the capability of the software product to be transferred from one environment to another.
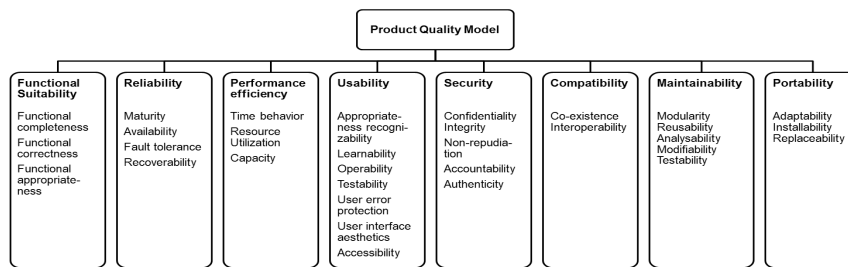
Fig. 1 ISO 25010:2011 - Software product quality

The quality in use model covers the outcome of the interaction when a product is used in a particular context of use providing a different perspective on quality. In our paper, we are interested in how quality in use is supported by MDE. The model is presented in figure 2 and is also specified in terms of quality attributes and sub-attributes. Again, we will use the quality attributes to guide our discussion on how quality is supported in MDE.

- *Effectiveness* is the capability to enable users to achieve specified goals with accuracy and completeness in a specified context of use.
- *Efficiency* is the capability to enable users to achieve specified goals in a timely manner, in a specified context of use.
- *Satisfaction* is the capability to satisfy users in a specified context of use.
- *Safety* is the capability to achieve acceptable levels of risk of harm to people, software, equipment or the environnent in a specified context of use.
- *Usability* is the capability to be easily used in a specified context of use.
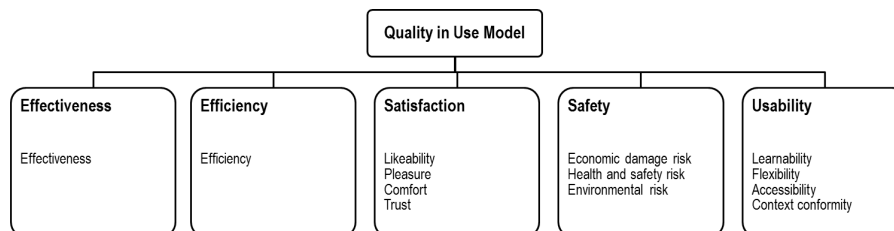
Fig. 2 ISO 25010:2011 - Quality in use
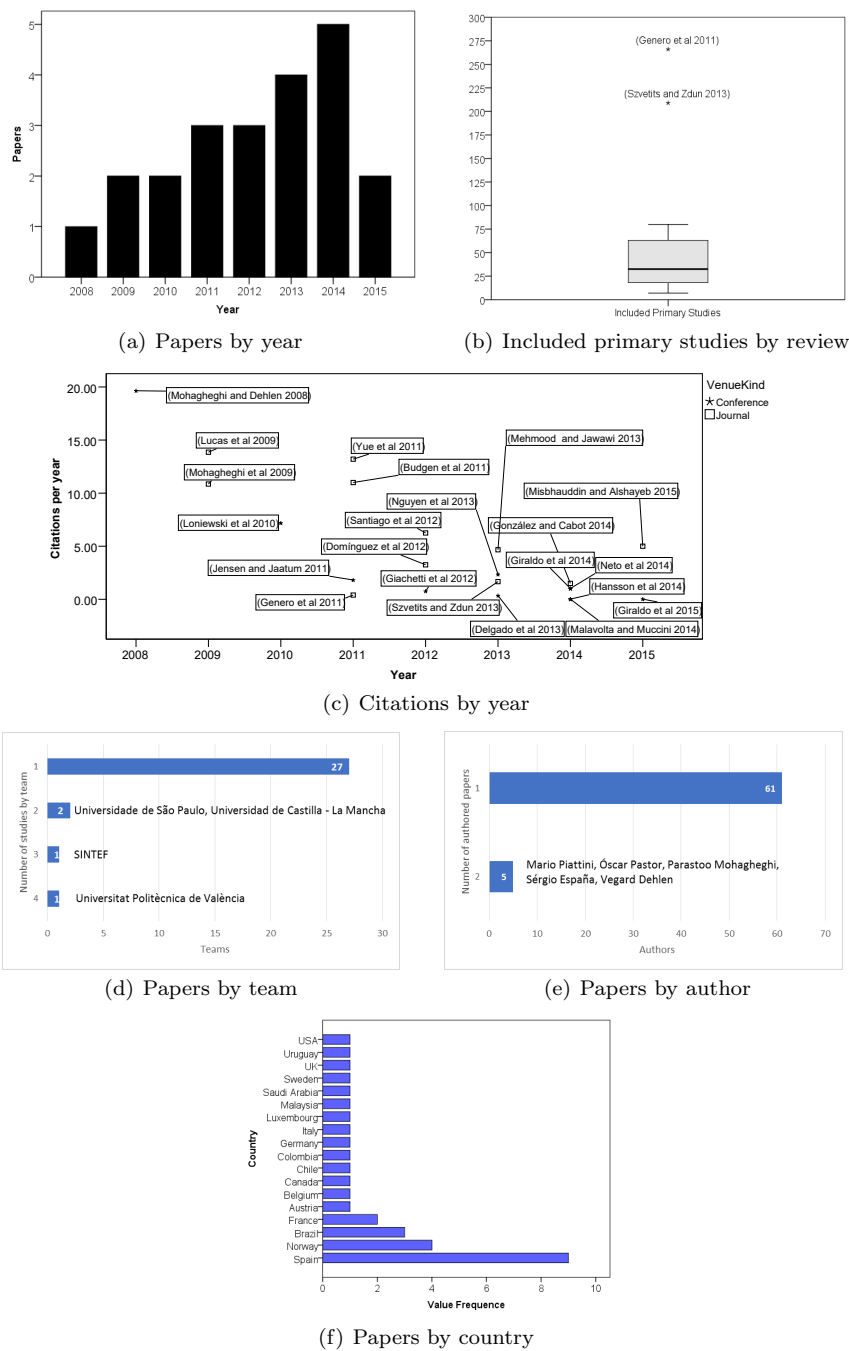
3 Results

3.1 Search results

We selected 22 papers for inclusion in this tertiary study. There is an overall increasing trend in the number of secondary studies published covering this topic,

which denotes a growing awareness on the subject (figure 3(a)). Note that the data was collected in mid 2015, which may explain why there is a relatively low number of publications in that year. The distribution of the number of included primary studies has a mean of 53, a standard deviation of 63.4 with a minimum of 7 and a maximum of 266 included primary studies, in a total of 1166 included primary studies (without considering repetitions among different citations lists). The box-plot diagram in figure 3(b) highlights two extremes, representing two secondary studies with a very high number of included primary studies, when compared to the other secondary studies.

The impact of the selected publications can be assessed through the assessment of the number of citations of each paper. As of July 16, 2015, the selected papers were cited 585 times, according to Google Scholar, with a mean of around 26.6 citations per paper, and a standard deviation of 40.7. Older papers have in general more citations than more recent ones. Figure 3(c) presents the number of citations per paper divided by the paper's age, in years, distributed by year of publication and encoded by publication type (journal vs. conference paper). We present the average, rather than the total number of citations per paper to mitigate the cumulative effect of the paper's age in its number of citations. Journal papers tend to have more impact than conference papers, in this sample. The selected paper's set *h-index* is 9. The *h-index* is the largest number h such that h publications have at least h citations (Hirsch 2005). The *h-index* is often used to characterize the scientific output of a researcher. In this case, we use it to assess the impact of the body of publications included in this tertiary study.

We also characterize the community performing these studies. The highest frequency of papers by institution is from Universitat Politècnica de València - Spain (4), SINTEF - Norway (3), Universidade de São Paulo - Brazil (2) and Universidad de Castilla - La Mancha - Spain (2). All the remaining institutions contribute with 1 paper (figure 3(d)). 5 out of 66 authors have co-authored 2 of the selected papers each. The remaining 61 authors have only participated in one of the selected secondary studies (figure 3(e)). The topic is widely distributed, with researchers from 4 continents, 18 countries and 32 institutions, and Spain as the most active country conducting secondary studies that cover quality in MDE (figure 3(f)). Overall, there is a clear predominance of European teams, followed by teams from South America. Concerning international collaboration among organizations from different countries, one paper (Genero et al 2011) has authors from organizations from 3 different countries, while 6 papers have authors from organizations from 2 different countries. The remaining papers are authored by members of organizations from a single country. Again, Spain is the country involved in more international collaborations, participating in 4 out of 7 international teams.

Table 3 summarizes the information on the intended *audience*, *aim* of the consolidated information, and *object of study* of the selected secondary studies. Concerning the papers' audiences, here is a clear predominance of papers targeted to *researchers* in the selected secondary studies. All the selected papers are targeted to an academic audience. 5 of these papers are also targeted at *practitioners*. We identified 3 types of *aim* of the information consolidated in the selected papers. *Decision support* papers are the least common, with only 3 occurrences in our selection. This contrasts with 17 papers consolidating information relevant for *knowledge support*. All the selected secondary studies have *scoping* value for the respective areas. In other words, they provide an overview of the research area, col-

(a) Papers by year

(b) Included primary studies by review

(c) Citations by year

(d) Papers by team

(e) Papers by author

(f) Papers by country

Fig. 3 Demographics informations on selected papers.

lecting research evidence on the specific topic they address. Finally, we identified 3 main objects of study considered in these papers. 20 of these papers consolidate quality information about *models*. 14 papers cover model transformations. 10 papers discuss the quality of MDE-related software processes.

Table 3 Audience, aim and object of study of included studies

Paper Id	Audience Researchers	Practitioners	Review aim Decision	Knowledge	Scoping	Object of study Models	Transformations	Processes
(Budgen et al 2011)	✓	✓		✓	✓	✓		
(Delgado et al 2013)	✓			✓	✓	✓	✓	✓
(Domínguez et al 2012)	✓	✓	✓	✓	✓	✓	✓	
(Genero et al 2011)	✓				✓	✓		
(Giachetti et al 2012)	✓			✓	✓	✓	✓	✓
(Giraldo et al 2014)	✓			✓	✓	✓		✓
(Giraldo et al 2015)	✓	✓			✓	✓	✓	✓
(González and Cabot 2014)	✓			✓	✓	✓		
(Hansson et al 2014)	✓	✓	✓	✓	✓			✓
(Jensen and Jaatun 2011)	✓			✓	✓	✓		
(Loniewski et al 2010)	✓			✓	✓	✓	✓	
(Lucas et al 2009)	✓			✓	✓	✓	✓	
(Malavolta and Muccini 2014)	✓			✓	✓	✓		
(Mehmood and Jawawi 2013)	✓				✓	✓	✓	✓
(Misbhauddin and Alshayeb 2015)	✓				✓	✓	✓	
(Mohagheghi and Dehlen 2008)	✓	✓		✓	✓	✓	✓	✓
(Mohagheghi et al 2009)	✓			✓	✓	✓		
(Neto et al 2014)	✓			✓	✓	✓	✓	✓
(Nguyen et al 2013)	✓				✓	✓	✓	✓
(Santiago et al 2012)	✓			✓	✓	✓	✓	✓
(Szvetits and Zdun 2013)	✓	✓		✓	✓	✓	✓	
(Yue et al 2011)	✓		✓	✓	✓		✓	
Total	22	6	3	17	22	20	14	10

3.2 Quality evaluation of secondary studies

We applied the DARE quality criteria introduced in section 2.4 to all the selected papers. For the purposes of these classifications, we count criteria which are, at least, partially fulfilled. Finally, we also present the sum of all these quality criteria scores. The DARE **Q**uality **C**riteria are as follows:

- **QC1: Were inclusion and exclusion criteria reported?** To answer this question, we use a three-level scale. If the inclusion and exclusion criteria are explicitly reported, we encode this as 1. If these criteria are implicit, but can be safely inferred, we encode this as partial, corresponding to a score of 0.5. Finally, if the inclusion and exclusion criteria are absent from the paper, we grade this with 0.
- **QC2: Was the search adequate?** To answer this question, we use a three-level scale. If the study reports on using 4 or more data repositories in its automated search process and an additional search mechanism, such as snowball search, we mark this as 1. If the study reports on using 3 or 4 data data repositories with no extra search strategies, we mark it as 0.5. Finally, if previous two conditions were not fulfilled we mark this as 0.
- **QC3: Were the included studies synthesized?** We use a two-level scale for this question, with 1 standing for "Yes", and 0 for "No".
- **QC4: Was the validity of the included studies synthesized?** We use a three level scale for this criterion. We assign 1 to studies where the authors explicitly defined quality or validity criteria and extracted them from the primary study. We assign 0.5 to studies where the research question involves quality or

validity issues that are addressed by the study. We assign 0 if there is no explicit quality or validity assessment of the papers included in the secondary study.

– **QC5: Are sufficient details about the individual studies presented?**
We use a three level scale for this criterion. We assign 1 to papers when we can trace relevant information presented in a secondary study back to the primary study where it was collected from. We assign 0.5 when only summary information is presented about the individual papers. For example, papers may be grouped into specific categories, but it may not be explicit which papers belong to which specific category. Finally, we assign 0 when the results of individual studies are not specified.

With only two exceptions (Genero et al 2011; Malavolta and Muccini 2014), all the other studies meet at least the simplified DARE criteria, which are better suited to evaluate SLRs than SMSs. We opted to keep these two papers in our selected secondary studies as both are, in our opinion[2], best characterized as SMSs, which partially explains their relatively lower score. Evaluation criteria such as the search adequacy and the quality evaluation of the included primary studies are less critical in SMSs. Overall, the quality of included secondary studies is very good.

Table 4 Included secondary studies quality assessment

Secondary study	QC1	QC2	QC3	QC4	QC5	Total	Simple DARE	Full DARE
(Budgen et al 2011)	1	1	1	0	1	4	1	1
(Delgado et al 2013)	0.5	0.5	1	0	1	3	1	1
(Domínguez et al 2012)	1	1	1	0	1	4	1	1
(Genero et al 2011)	1	0.5	0	0	0	1.5	0	0
(Giachetti et al 2012)	1	1	1	1	1	5	1	1
(Giraldo et al 2014)	1	1	1	1	1	5	1	1
(Giraldo et al 2015)	1	0.5	1	0	1	3.5	1	1
(González and Cabot 2014)	1	1	1	0.5	1	4.5	1	1
(Hansson et al 2014)	0.5	0.5	1	1	1	4	1	1
(Jensen and Jaatun 2011)	1	0.5	1	0.5	1	4	1	1
(Loniewski et al 2010)	1	1	0	1	1	4	1	1
(Lucas et al 2009)	1	1	1	0	1	4	1	1
(Malavolta and Muccini 2014)	1	0	1	0	1	3	0	0
(Mehmood and Jawawi 2013)	1	1	1	0	1	4	1	1
(Misbhauddin and Alshayeb 2015)	1	0.5	1	1	1	4.5	1	1
(Mohagheghi and Dehlen 2008)	0.5	0.5	1	0	1	3	1	1
(Mohagheghi et al 2009)	1	1	1	0	1	4	1	1
(Neto et al 2014)	1	0.5	1	1	1	4.5	1	1
(Nguyen et al 2013)	1	1	1	0	0	3	1	0
(Santiago et al 2012)	1	1	1	1	1	5	1	1
(Szvetits and Zdun 2013)	1	0.5	1	0	1	3.5	1	1
(Yue et al 2011)	1	1	1	1	1	5	1	1

All the selected secondary studies were built using at least one guideline, or a combination of guidelines. (Kitchenham and Charters 2007) is the most frequently used set of guidelines, similar with what we find in recent secondary and tertiary studies. A particularly noteworthy exception to the rule is the primary studies identification strategy followed in (Giraldo et al 2015). Rather than essentially using a search-based approach to detecting primary studies to be potentially included, which may then be complemented with other strategies, as suggested in

---

[2] Although (Genero et al 2011) was published as a SLR, its 5 research questions are of a mapping nature. Under these circumstances, the quality classifications, which can be seen as *adequacy for inclusion* classifications can be less strict.

(Kitchenham and Charters 2007), (Giraldo et al 2015) use the backward snowball approach, which was recently evaluated as an effective alternative for performing literature reviews in (Wohlin 2014).

Table 5 Guidelines used in the secondary study

Paper Id	(Biolchini et al 2005)	(Brereton et al 2007)	(Kitchenham et al 2004)	(Kitchenham and Charters 2007)	(Kitchenham et al 2009)	(Mohagheghi and Conradi 2007)	(Petersen et al 2008)	(Petticrew and Roberts 2008)	(Wohlin 2014)
(Budgen et al 2011)				✓					
(Delgado et al 2013)			✓	✓					
(Domínguez et al 2012)			✓	✓					
(Genero et al 2011)		✓	✓	✓					
(Giachetti et al 2012)				✓				✓	
(Giraldo et al 2014)				✓					
(Giraldo et al 2015)									✓
(González and Cabot 2014)			✓	✓					
(Hansson et al 2014)				✓					
(Jensen and Jaatun 2011)			✓						
(Loniewski et al 2010)			✓						
(Lucas et al 2009)				✓					
(Malavolta and Muccini 2014)				✓					
(Mehmood and Jawawi 2013)				✓			✓		
(Misbhauddin and Alshayeb 2015)				✓					
(Mohagheghi and Dehlen 2008)						✓			
(Mohagheghi et al 2009)				✓					
(Neto et al 2014)				✓					
(Nguyen et al 2013)				✓					
(Santiago et al 2012)	✓			✓					
(Szvetits and Zdun 2013)				✓	✓				
(Yue et al 2011)				✓					
Totals	1	1	6	18	1	1	1	1	1

All but (Nguyen et al 2013) make the list of included primary studies available, either directly on the paper, or through an online resource (e.g. a companion site for the paper, or a more detailed technical report). 9 secondary studies explicitly assess the quality of the primary studies they include. 8 of those secondary studies use an *ad-hoc* assessment, with custom-made criteria tailored for the specific domain those reviews are covering (Delgado et al 2013; Domínguez et al 2012; Giachetti et al 2012; Giraldo et al 2014; Hansson et al 2014; Misbhauddin and Alshayeb 2015;

Mohagheghi and Dehlen 2008; Santiago et al 2012). (Neto et al 2014) uses guidelines (Kitchenham and Charters 2007) for the quality assessment of the included primary studies.

3.3 Findings

3.3.1 On the nature of the research questions

Each secondary study has its own set of research questions. We identify two major kinds of research questions in these papers: those which are primarily focused in analysing trends in a particular quality in MDE toping area (mapping questions), and those aggregating data from different studies, typically to compare several alternatives, according to a specific set of quality attributes (aggregation questions). 80 of the research questions on the selected secondary studies have essentially a mapping nature (figure 4). Only 3 research questions are of the form *"How does A compare to B?"*, where *A* and *B* are MDE tools, techniques, processes, etc.
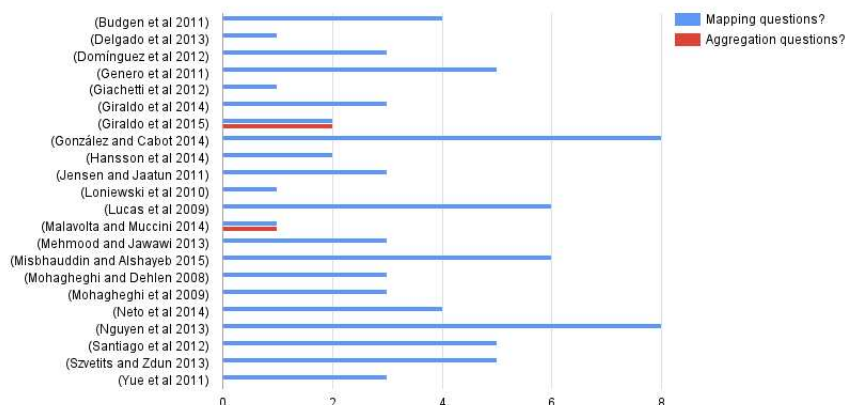
Fig. 4 Research questions categorization

3.3.2 On the coverage of quality attributes by the secondary studies

Table 6 summarizes the *ISO 25010:2011 - Software product quality* attributes, and their coverage in the secondary studies included in this review. All the papers address at least one of the quality attributes. The most commonly addressed quality attribute is maintainability with 18 out of the 22 secondary studies. All

the remaining quality attributes are addressed by a relatively low number of primary studies with 5 papers addressing reliability, efficiency and usability, 4 papers covering security, compatibility and portability, and 3 papers covering functional suitability.

Table 6 Coverage of ISO 25010:2011 - Software product quality

Paper Id	Product Quality	Functional Suitability	Reliability	Efficiency	Usability	Security	Compatibility	Maintainability	Portability
(Budgen et al 2011)	✓							✓	
(Delgado et al 2013)	✓		✓					✓	
(Domínguez et al 2012)	✓			✓	✓			✓	
(Genero et al 2011)	✓	✓	✓					✓	
(Giachetti et al 2012)	✓						✓	✓	
(Giraldo et al 2014)	✓							✓	
(Giraldo et al 2015)	✓		✓	✓	✓		✓	✓	✓
(González and Cabot 2014)	✓							✓	
(Hansson et al 2014)	✓							✓	
(Jensen and Jaatun 2011)	✓					✓			
(Loniewski et al 2010)	✓							✓	
(Lucas et al 2009)	✓							✓	
(Malavolta and Muccini 2014)	✓			✓				✓	
(Mehmood and Jawawi 2013)	✓							✓	
(Misbhauddin and Alshayeb 2015)	✓							✓	
(Mohagheghi and Dehlen 2008)	✓							✓	✓
(Mohagheghi et al 2009)	✓	✓			✓			✓	
(Neto et al 2014)	✓		✓		✓	✓	✓		✓
(Nguyen et al 2013)	✓					✓			
(Santiago et al 2012)	✓							✓	
(Szvetits and Zdun 2013)	✓		✓	✓	✓	✓	✓	✓	✓
(Yue et al 2011)	✓	✓		✓					
Frequency	22	3	5	5	5	4	4	18	4

Table 7 summarizes the coverage of *quality in use* attributes in the selected secondary studies. The quality in use is clearly a less explored view of quality, when compared to the product quality model. The model includes 5 attributes, of which the most explored is usability, with 7 occurrences. Effectiveness was explored in 2 studies. Finally, efficiency, satisfaction, and safety are the least explored quality in use attributes, being covered by 2 papers each. 9 papers do not cover quality in use at all.

Table 7 Coverage of ISO 25010:2011 - Quality in Use

Paper Id	Quality in use	Effectiveness	Efficiency	Satisfaction	Safety	Usability
(Budgen et al 2011)	✓					✓
(Delgado et al 2013)						
(Domínguez et al 2012)	✓					✓
(Genero et al 2011)						
(Giachetti et al 2012)						
(Giraldo et al 2014)	✓			✓		✓
(Giraldo et al 2015)	✓	✓		✓		✓
(González and Cabot 2014)						
(Hansson et al 2014)	✓	✓				
(Jensen and Jaatun 2011)						
(Loniewski et al 2010)	✓		✓			
(Lucas et al 2009)	✓	✓	✓			
(Malavolta and Muccini 2014)	✓	✓				
(Mehmood and Jawawi 2013)						
(Misbhauddin and Alshayeb 2015)	✓					
(Mohagheghi and Dehlen 2008)						
(Mohagheghi et al 2009)	✓					✓
(Neto et al 2014)	✓				✓	✓
(Nguyen et al 2013)						
(Santiago et al 2012)						
(Szvetits and Zdun 2013)	✓				✓	✓
(Yue et al 2011)	✓					
Frequency	13	4	2	2	2	7

3.4 Summary of secondary studies contributions

In this section, we briefly outline the main contributions of each of the included secondary studies. In some situations, where there is a strong connection between more than one SLR (or SMS), we merge their discussion, for easier comparison. For example, (Budgen et al 2011; Genero et al 2011) are both about UML models; (Giraldo et al 2014, 2015) are closely related in that they explore the concept of quality, first in MDE in general, and then in MDE languages in particular.

(Budgen et al 2011) presents a SLR on how the UML has been empirically evaluated. The authors concluded that *comprehension* was the most widely studied category (with impacts on the maintainability and usability of the language), and that *adoption* was an emerging topic which would interest practitioners. More generally, all topics could benefit from further research, particularly because there was a strong preponderance of lab evaluations, frequently performed with students, with relatively short durations, and a corresponding lack of field studies. Moreover, although UML has several different diagrams, there was a very strong predominance of studies covering class diagrams. Conversely, the remaining diagrams are insufficiently studied. An important conclusion of the SLR was that there was a clear need for more works to question its fitness for the intended purpose, rather than taking it for granted. (Genero et al 2011) is also dedicated to the quality of UML models. In this case, the authors focus on conceptual models. The overall conclusions are compatible with those of (Budgen et al 2011). In particular, the authors emphasize the need for much more empirical validation of claims concerning UML conceptual models, and a much stronger interaction with practitioners in that validation. Most of the identified studies focus on semantic quality, but very few cover semantic completeness.

The methodological or conceptual applications of service-oriented computing and development, as well as **M**odel-**D**riven **D**evelopment (MDD) and MDE, to business processes and business process management are explored in an SLR reported in (Delgado et al 2013). The SLR discusses the integration issues among these paradigms contrasting the studies which integrate business processes with model driven approaches, with service oriented approaches, and a combination of model-driven and service-oriented approaches. The latter is clearly the most commonly explored integration approach. The review discusses impacts of this integration which suggest positive effects on reliability and maintainability.

State based languages, namely UML state machines, finite state machines and Harel state charts, can be used as a source for generating code. (Domínguez et al 2012) presents an SLR where some of the reviewed proposals discuss the positive impact on quality attributes such as reusability and maintainability. Most of the works build upon software design patterns. As an identified shortcoming, it is often the case that several proposals do not fully support important elements of rich state machine specifications. Another common shortcoming is that several proposals fail to provide an implementation strategy considering qualitative aspects in software development.

The multitude of MDD approaches and languages creates important challenges in terms of interoperability. (Giachetti et al 2012) surveyed this research area and found that although this was a hot topic, it was common to find several different approaches tackling essentially the same issues, but not relating them, particularly when it came to interoperability approaches focusing on supporting MDD

processes. The authors found challenges in reaching a consensus in terminology and concepts used, which is a crucial step towards the desired interoperability.

(Giraldo et al 2014) discuss how the concept of quality is tackled in MDE literature. They identify a plethora of quality in MDE definitions (and some misconceptions, as well) and their associated trends, resulting from the different perspectives each of the primary studies is supporting. In an extension to this work, a second review is dedicated to the proliferation of modeling languages, and the quality of languages and models built with those languages. (Giraldo et al 2015) explore this topic and propose a research agenda, in order to cover several existing gaps in the current state of the art in modeling languages quality evaluation. In particular, they point to the growing complexity involved in MDE (in this case of an accidental nature, due to insufficient tool support, on the one hand, and to the lack of adequately trained personnel, on the other), the problem of dealing with several different complementary languages which may be used at the same, or different abstraction levels, and the importance of considering model transformations, as well.

When models are used as a starting point for a (semi-)automatic generation of the implementation, verifying them becomes crucial, as problems in the models can lead to undesired effects on the generated systems. (González and Cabot 2014) report on the existing approaches to formal verification of static MDD models. The approaches to verification in MDD typically start with a formalization step and are then followed by using the verification mechanisms available to the specific formalism used. This formalization is often materialized in UML class diagrams (or some other similar approach), complemented with **O**bject **C**onstraint **L**anguage (OCL). The completeness of these approaches is strongly influenced by the level of support to OCL in the used tools. The most commonly verified properties are satisfiability and relationships among constraints (e.g. non-redundancy). The authors suggest the creation of a catalog of commonly agreed verification properties, and a set of benchmarks with which the approaches claiming to support their verification can be checked. This review highlights the need for a stronger efficiency and a better integration of the verification tools on the development tool chain, facilitating a more widespread adoption.

The SLR on model-driven agile development (Hansson et al 2014) concludes that this research field is still immature with respect to empirical evidence and industrial experience. The main aim of integrating agile practices into MDD were involving stakeholders, improving productivity and quality, and shortening the lead time. Due to immaturity the successfulness of the approach is still not yet quantified.

The main findings of an SLR on security in MDD (Jensen and Jaatun 2011) are based mainly on experience reports involving primarily prototypes. These primary studies provide little evidence that automatically generated code is more secure or better than those obtained by other development approaches. On the other hand it was reported that security design models are understandable and expressive enough to model different access control policies. The study also reveals that security experts must be able to evaluate the quality of transformation rules, which is often beyond their capabilities.

Some of interesting findings of SLR on the use of requirements engineering techniques in MDD are that about of 60% of primary studies included in (Loniewski et al 2010) already used models as a means to represent requirements,

and about 40% of them perform fully automatic transition from requirements specifications to analysis and design. Interestingly, almost 60% of included primary studies have some automated support for traceability management. However, a complete solution to manage models in requirements phase is still lacking. Furthermore, empirical studies showing enhanced productivity, efficiency and effectiveness are still missing.

Models are frequently specified with different languages, each providing its specialized view on the system. UML, for example, is composed of several different sub-languages, and the different model views provided by each language must be consistent among them. (Lucas et al 2009) reviews the state of the art of UML model consistency checking. One of the important outcomes is that vertical inconsistency problems were (at least back in 2009) much less explored by research than horizontal ones. The authors of the review also proposed an approach to mitigate the identified gaps in the existing research, building on models transformations and rewriting logic.

(Malavolta and Muccini 2014) reports a SMS on how MDE can be used in the context of on wireless sensor networks. The authors identify the main motivations for researchers to use MDE in this research field. The main conclusion was that MDE is used due to automatic support of code generation and documentation, as well as support for different analysis (e.g., performance, fault tolerance, power consumption, security). Both have impact on different quality issues. However, there is still lack of a satisfactory standard language for modeling wireless sensor networks.

The key findings of a SMS on aspect-oriented model-driven code generation by (Mehmood and Jawawi 2013) are: although the most significant advantages of automatic model-driven code generation are reduction of development time and improvement in maintainability, extensibility, and reliability these have yet to be proved for aspect-oriented model-driven modeling since mainly solution proposals exist. Validation and evaluation of these proposal are rare indicating that model verification is harder since designers need to know the details of advice transformations resulting in usability problems; aspect-oriented features are less appropriate for complex modeling situations that require weaving beyond simple model composition and transformation (e.g., with complex join-points); and problems with dynamic aspect weaving and unweaving.

The effect of model refactoring on model quality has been one of the research questions in SLR (Misbhauddin and Alshayeb 2015). It turns out that only 5 primary studies out of 63 addressed this topic and research is clearly inadequate. The only technique applied is to use quality design metrics before and after refactoring. However, established correlation between these metrics and quality attributes is still missing.

Ultimately, the adoption of MDE in industry should be guided by a clear understanding of the involved benefits and limitations. (Mohagheghi and Dehlen 2008) reviewed publications from 2000 to 2007 covering industrial experiences with MDE and found that, at the time, third-party tool maturity was not satisfactory for large-scale adoption. The authors did find some evidence on the benefits of MDE adoption in industry, where improvements in software quality were accompanied with reports of productivity gains and losses. However, these reports were mostly based on small-scale projects, so studies on larger projects were neces-

sary to strengthen the evidence. Overall, the gathered evidence was insufficient for generalization of results, particularly for large-scale projects.

(Mohagheghi et al 2009) presents a SLR discussing the meaning of model quality and how to improve it. In this work, six model quality goals were identified: correctness, completeness, consistency, comprehensibility, confinement and changeability. Some extra findings are discussed such as, for instance, to manage changeability and complexity of large and complex models, keep them consistent, and verify quality on the model level are challenges in model-driven engineering that are not yet properly covered.

(Neto et al 2014) addresses the usage of MDE in the development of **S**ystems **of S**ystems (SoS). The authors found that UML and OCL are the most frequently reported languages used in this context. Eclipse-based tools built with EMF/GMF are the most commonly reported as supporting the construction of SoS. Although there is a wide coverage of quality attributes in this context, interoperability is the most frequent quality concern, followed by reliability, safety and security.

In the SMS presented in (Nguyen et al 2013) we can find a taxonomy focused on specialized Model-Driven Engineering approaches for supporting the development of systems regarding security (also called Model-Driven Security, MDS). This study shows that authorization, especially access control, and confidentiality are the security concerns that are mostly addressed by MDS. Regarding the modeling approaches, the separation of concerns is regarded as prevalent, and **D**omain-**S**pecific **L**anguages (DSLs) are a popular approach for leveraging MDE for secure systems development (despite the fact that the great majority of those DSLs are implemented using UML profiles and stereotypes). Also, a small fraction of those describe the semantics properly, which means that they are not meant for formal analysis. Finally, a residual part of the selected studies present some sort of evaluation mostly validating via illustrations.

(Santiago et al 2012) focuses on traceability management in the context of MDE. They observe that the most addressed traceability operations are storage, **c**reate, **r**ead, **u**pdate and **d**elete (CRUD), and visualization. However, often CRUD operations are only partially supported, and the visualization mechanisms are typically ad-hoc and do not fit well the specific needs of visualizing traceability information, according to the author's analysis. Another identified shortcoming is the predominance of technology dependent traceability links generation from model transformations which hampers their application to other existing model transformation languages. The traceability metamodel is also proposal-specific, which complicates information interchange.

Models at runtime cope with the systems dynamic aspects. (Szvetits and Zdun 2013) reviews and classifies the current research approaches with respect to their objectives (e.g. adaptation, abstraction, consistency, conformance, error handling, monitoring, simulation, prediction, platform independence, and policy checking and enforcement), techniques (e.g. introspection, model conformance, model comparison, model transformation, and model execution), architectures (e.g. monolithic, local dataflow, middleware, repository, or a combination of several of these), and kinds of models used in these approaches. A common trend observed in the included primary studies is a concern shift from low-level system interactions to model-based processing, which is closer to the problem domain.

Focused on transformation approaches between user requirements and analysis models, paper (Yue et al 2011) is a SLR that unveils that there is still not prac-

tical, workable and automated solution in this topic. Those existing ones require significant user effort to document requirements, or are inefficiency, or are not able to (semi-) automatically generate a complete and consistent analysis model. The authors suggest to keep researching on transformations focusing on empirical studies, their quality characteristics such as usability, efficiency, scalability, extensibility, and interoperability.

4 Discussion

In this section we revisit the research questions, in light of the data gathered in this tertiary review (sub-section 4.1). This discussion follows with the proposal of a research roadmap, in section 4.2. Finally, we discuss the limitations of this study and our mitigation strategies for each of them in section 4.3.

4.1 Discussion of the Research Questions

RQ1: What is the currently available information concerning quality in MDE, systematically aggregated by means of a SLR, or a SMS?
There is already a relatively large body of work synthesized which is somewhat related to quality in MDE. Our review included 22 secondary studies with a mean number of 53 included primary studies. Overall, the selected secondary studies included a total of 1166 primary studies (RQ1.1). The primary studies span from 1990 to 2014 (RQ1.2), and their list is available for all but one of the included secondary studies (RQ1.3). However, in most cases, the quality of the primary studies is not explicitly assessed (RQ1.4). When this assessment is performed, it is mostly done *ad-hoc*, with custom-made criteria for this assessment. In some cases, the assessment of the primary studies is performed following suggestions in (Kitchenham and Charters 2007).

The available secondary studies are mostly targeted to researchers – all the selected secondary studies have researchers as their target audience (RQ1.5). Nevertheless, 6 of the secondary studies are also aimed at practitioners. This large predominance of research-oriented papers may be interpreted as a hint for the level of maturity of the research domains covered in the reviews, which may be perceived as mature enough so that a consolidation effort is necessary, but perhaps not yet at a point where this consolidation is at the adequate level for discussing the collected evidence with practitioners. An alternative explanation could be that researchers may not necessarily regard these secondary studies as the best vehicle for conveying the consolidated findings to practitioners. Further research on this topic is certainly necessary. This shortcoming of lack of advice targeted for practitioners is not exactly surprising. In a tertiary study on SLRs and SMSs in software engineering a similar problem was identified (Kitchenham et al 2009). The original purpose of the Evidence-Based Software Engineering movement was to provide consolidated evidence to practitioners, as well, so more research targeting practitioners is necessary. Whatever the reason is, we note that although all the included secondary studies have a scoping goal, and most of them are also concerned with knowledge consolidation, only three of those studies explicitly address decision support as one of their aims. Again, this may be a symptom of an

insufficient orientation for practitioners of the results in these secondary studies (RQ1.6).

Different aspects of MDE (and how quality relates to them) are covered in these secondary studies (RQ1.7): 20 of the included secondary reviews cover models; 14 cover transformations; finally, 10 cover processes. Although quality models are not explicitly discussed in the selected secondary studies, we found that all those studies discuss at least one aspect of software product quality (RQ1.8). The most frequently addressed quality attribute is maintainability. This is in line with one of the "selling points" of MDE: the potential economic benefits it may bring in terms of costs savings in the software process. Each of the remaining product quality attributes listed in the ISO/IEC 25010:2011 standard is explored in a few secondary studies (from 3 to 5, depending on the specific quality attribute). We also explored the ISO/IEC 25010:2011 standard quality in use attributes, and found usability to be the most frequently covered (in 7 of the 22 secondary studies). This has mostly to do with the readability and understandability of the models. In other words, the added value of the MDE approach in terms of communication (e.g. by increasing the understandability of the software product through models representing it) is the most explored quality in use feature. Again, this has a potential impact on the economics of MDE, and in particular, in the maintainability of software developed with MDE.

RQ2: What is the current status of consolidation of data collected from different literature reviews covering quality in MDE?

As can be seen from section 3.4, there is little overlap in the focus of the different included secondary studies. In other words, the intersection of the object of study and targeted quality attributes in the different primary studies is nearly empty. This makes performing meta-analysis focusing on a particular niche of this domain harder, at a tertiary level. At the secondary level (the one of the included studies) we note that the research questions are predominantly of a "mapping nature". Out of 83 research questions collected overall, only 3 were not essentially mapping questions. Mapping questions are, of course, extremely important, as they help locating relevant research about each of the topics addressed in the secondary studies. However, the relative absence of questions of the form *"How does A compare to B?"*, where *A* and *B* are MDE tools, techniques, processes, etc, denotes a relatively low level of adoption of meta-analysis. This is not uncommon in Software Engineering, in general. As a community, we still struggle to aggregate independently collected evidence. Although the Evidence-Based Software Engineering approach is becoming more widespread, its adoption is still relatively low when compared to other domains. The same general problem is observable in this particular context of quality in MDE. This shortcoming of the current state of the art is, in our opinion, a *"growing pain"* in a maturing research area and should be regarded as an opportunity for much needed further research.

RQ3: Who are the key players in consolidating knowledge on quality in MDE through literature surveys?

There is a wide dispersion of authors performing these literature reviews, with only 5 out of 66 authoring more than one secondary study, and even those only authored two studies. In other words, there does not seem to be a well defined set of key players contributing to the consolidation of knowledge in quality in MDE, following this particular approach. Obviously, this is not likely to be the case in

each software quality niche, as explored by the corresponding primary studies. We found 22 eligible secondary studies (RQ3.1) addressing the topic of quality in MDE. This is a highly significant number of secondary studies. For comparison, a comprehensive tertiary study (and with a larger scope) on how research synthesis is performed in software engineering, included 49 secondary studies published from 2005 to 2010 (Cruzes and Dybå 2011). This is partly explained by the increasing adoption of secondary studies in software engineering, but also reflects a vibrant search area, with respect to the topic of our paper. These studies are mostly conducted by non-intersecting teams spread around 4 continents, 18 countries and 32 institutions (RQ3.2). Overall, and perhaps unsurprisingly, rather than identifying a community specialized in conducting these reviews for this particular domain, we found a more *"opportunistic"* community, in the sense that these teams spotted gaps in the consolidated knowledge on quality in MDE, in the scope of their research work, and took the opportunity to fulfill them with much needed secondary studies. This hints to the increasingly widespread adoption of this kind of reviews, which has clearly outgrown, as intended, the specialized Evidence-Based Software Engineering community.

In terms of impact (RQ3.3), the selected publications have an *h-index* of 9. Considering some of the included publications are quite recent, there is clearly an audience for these secondary studies.

4.2 A research roadmap for Quality in MDE

As stated in Introduction the main aim of this tertiary study is collecting the evidence concerning the impact of MDE in quality. In particular, we were interested in the quality of products build with MDE and the quality of process using MDE. In this respect 22 secondary studies, SLRs and SMSs, have been surveyed to get an overview of how MDE impacts quality and to provide a roadmap for future research of quality in MDE. Although a substantial body of primary and secondary study exists (22 secondary studies which examined over 1000 primary studies) none of the secondary studies explicitly addresses quality attributes for product quality and quality in use. There is a clear **lack of secondary studies collecting evidence of possible enhanced quality brought by MDE**.

Industry needs a clear evidence on MDE impacts on quality supported by controlled experiments in small, medium, and large environments. Findings from such primary studies then need to be synthesized by secondary studies providing a clear evidence for industry. This current lack of orientation towards practical use is very visible from our study. **All 22 secondary studies used in our tertiary study have been intended to use for researchers, only a few of them addressed also practitioners.**

We urge prospective researchers to **explicitly address quality attributes** in their secondary studies. Maintainability is among those software product quality attributes, which were most often implicitly addressed. Much less attention has been paid to other software product quality attributes, such as functional suitability, compatibility, performance efficiency, security, compatibility, and portability. Although some secondary studies provide a weak evidence on positive effects of MDE on maintainability, reusability and reliability, we need studies which will address these attributes explicitly.

On the other hand, **quality in use attributes were less investigated than software product quality attributes** (all 22 secondary studies implicitly address at least one software product quality attributes, whilst only 13 secondary studies implicitly addressed quality in use attributes). There is a clear lack of primary studies (controlled experiments) and secondary studies, which will explicitly address and synthesize knowledge about quality in use attributes, such as: effectiveness, efficiency, satisfaction, safety, and usability. Industry needs evidence supported by secondary studies which will explicitly cover the impact of MDE on quality in use attributes.

Last, but not least, our tertiary study clearly shows that **most of secondary studies consolidate quality information about models, much less about model transformations and software processes**. These gaps should be addressed in the near future. We encourage MDE researchers to start addressing the identified gaps to support practitioners with evidence on MDE impacts on quality.

4.3 Limitations of this study

A first potential limitation of this study is the coverage of our search and selection process. It is possible that a relevant secondary study may have been missed, either for not being captured at all with our search strings, in the chosen repositories, or for being mis-classified as irrelevant, when in fact it was relevant. The fact that our initial set of identified studies was all captured by our search strings gives us some confidence on the coverage of the search strings, although we did identify 4 extra studies (1 of which was eventually discarded), not captured through the search strings, using snowball search. As for the potential for missing a paper within the set, it is always possible that an unclear, or mis-interpreted, title and abstract could lead to a wrong exclusion. To mitigate this threat, we were very conservative in our exclusion decisions. When in doubt, we kept the papers for further analysis by a different author. If the doubt persisted, the paper passed on to the next phase.

A second threat, common to this kind of studies, is that there is always some potential for mis-interpreting the secondary study. Again, we tried to be conservative, used a common data collection tool (implemented as a shared spreadsheet), and when in doubt marked the specific data element for discussion with the remaining authors, so that a consensus interpretation could be reached.

Finally, the authors of the included secondary studies may have also had problems both in the selection and the interpretation of the primary studies. If wrong conclusions were reached in an included secondary study, these could propagate to our tertiary study. To gauge this threat, we used the DARE Quality Criteria to assess the quality of the included secondary studies. The included secondary studies were of high quality, giving us some confidence that this potential propagation of wrong conclusions did not occur. Also, we did not perform meta-analysis on the data collected from the secondary studies, which would have the potential effect of amplifying errors from the secondary studies included in this tertiary study.

5 Conclusions and further work

This paper contributes with a tertiary study on the quality in MDE. We identified 22 secondary studies (SLRs and SMSs). In these secondary studies, we looked for discussions on how MDE impacts different quality attributes, taken from the ISO standards quality attributes for product quality and for quality in use. The product quality was chosen so that we could analyse existing work in a wide range of internal and external product quality attributes. The quality in use was chosen for this tertiary study as we were also interested in learning about the current body of knowledge of the impact of MDE in the quality in use of products built with it. We now revisit the main contributions of this paper.

The first contribution concerns mapping the most representative secondary studies that cover quality in MDE, their origin (to identify key players in this research area), and the quality attributes addressed by each of the secondary studies. The 22 studies mostly come from independent teams, each concerned with its own niche. The coverage of quality attributes is wide, but with a predominance of product quality attributes over quality in use attributes. This observation seems to be inline with the common claims of MDE having a positive impact in terms of costs reduction in software development, and not so much in the quality of the product as perceived by its end user, who typically does not even need to know how the product was built. Maintainability is, by far, the most often addressed quality attribute. In general the contributions are more often targeted to researchers than to practitioners, which are only clearly targeted by 6 of the reviews.

A second contribution of this paper is an annotated overview of the existing aggregated information on quality in MDE. This is intended to serve as a starting point for interested readers to explore particular niches, by directing them to a particular literature review.

The third contribution of this paper is an analysis on the level of consolidation of the aggregated information on quality in MDE. The large predominance of questions more aimed at mapping the current body of knowledge than to compare among two, or more, alternatives, is a symptom of the relative novelty of the area (when compared to more "traditional" sciences), which is reaching a maturity level where the secondary studies can progressively focus more on answering more specific research questions comparing among different alternatives. The wide dispersion of sub-topics addressed in the different reviews makes it hard to perform meta-analysis on the collected data, hampering some of the potential benefits of SLRs.

Last, but not the least, we propose a research roadmap for Quality in MDE, based on the main findings of our tertiary study. The identified shortcomings of the current state of the art in this domain offer important opportunities for conducting relevant consolidation efforts and, consequently, pushing MDE envelope to practitioners who still may have doubts on the potential benefits for their specific contexts.

References

Agner LTW, Soares IW, Stadzisz PC, Simão JM (2013) A Brazilian survey on UML and model-driven practices for embedded software development. Journal of Systems and Software 86(4):997–1005, DOI 10.1016/j.jss.2012.11.023

Ameller D, Franch X, Gómez C, Araujo J, Svensson RB, Biffl S, Cabot J, Cortellessa V, Daneva M, Fernández DM, Moreira A, Muccini H, Vallecillo A, Wimmer M, Amaral V, Brunelièrek H, Burgueño L, Goulão M, Schätz B, Teufl S (2015) Handling Non-Functional Requirements in Model-Driven Development: An Ongoing Industrial Survey. In: 23rd IEEE International Requirements Engineering Conference (RE 2015)

Badreddin O, Lethbridge TC, Elassar M (2013) Modeling practices in Open Source Software. In: Open Source Software: Quality Verification, Springer, pp 127–139, DOI 10.1007/978-3-642-38928-3_9

Badreddin O, Sturm A, Hamou-Lhadj A, Lethbridge T, Dixon W, Simmons R (2015) The effects of education on students' perception of modeling in Software Engineering. In: First International Workshop on Human Factors in Modeling (HuFaMo 2015), CEUR Workshop Proceedings

Biolchini J, Mian PG, Natali ACC, Travassos GH (2005) Systematic Review in Software Engineering. Tech. Rep. RT–ES 679 / 05, System Engineering and Computer Science Department COPPE/UFRJ, URL `http://www.cin.ufpe.br/~in1037/leitura/systematicReviewSE-COPPE.pdf`

Brereton P, Kitchenham BA, Budgen D, Turner M, Khalil M (2007) Lessons from applying the Systematic Literature Review process within the Software Engineering domain. Journal of Systems and Software 80(4):571–583, DOI 10.1016/j.jss.2006.07.009

Budgen D, Kitchenham BA, Charters SM, Turner M, Brereton P, Linkman SG (2008) Presenting Software Engineering results using structured abstracts: a randomised experiment. Empirical Software Engineering 13(4):435–468, DOI 10.1007/s10664-008-9075-7

Cruzes DS, Dybå T (2011) Research synthesis in software engineering: A tertiary study. Information and Software Technology 53(5):440–455, DOI 10.1016/j.infsof.2011.01.004

Cuadrado JS, Molina JG, Tortosa MM (2006) Rubytl: a practical, extensible transformation language. In: Model Driven Architecture–Foundations and Applications, Springer, pp 158–172

Cuadrado JS, Izquierdo JLC, Molina JG (2014) Applying Model-Driven Engineering in small software enterprises. Science of Computer Programming 89:176–198, DOI 10.1016/j.scico.2013.04.007

Dybå T, Dingsøyr T (2008) Strength of evidence in systematic reviews in software engineering. In: Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, ACM, pp 178–187, DOI 10.1145/1414004.1414034

Fernández-Sáez AM, Genero M, Caivano D, Chaudron MRV (2015) On the use of UML documentation in software maintenance: Results from a survey in industry. In: ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS 2015), ACM/IEEE, pp 292–301

Forward A, Lethbridge T, Badreddin O (2010) Problems and opportunities for model-centric vs code-centric development: A survey of software professionals. In: 5th Workshop "From code centric to model centric: Evaluating the effectiveness of MDD (C2M: EEMDD)", University Pierre & Marie Curie, Paris

Hirsch JE (2005) An index to quantify an individual's scientific research output. Proceedings of the National Academy of Sciences of the United States of America 102(46):16,569–16,572, DOI 10.1073/pnas.0507655102

Hutchinson J, Whittle J, Rouncefield M, Kristoffersen S (2011) Empirical assessment of MDE in industry. In: Proceedings of the 33rd International Conference on Software Engineering, ACM, New York, NY, USA, ICSE '11, pp 471–480, DOI 10.1145/1985793.1985858

Hutchinson J, Whittle J, Rouncefield M (2014) Model-Driven Engineering practices in industry: social, organizational and managerial factors that lead to success or failure. Science of Computer Programming 89:144–161, DOI 10.1016/j.scico.2013.03.017

ISO/IEC (2011) Systems and software engineering – systems and software quality requirements and evaluation (square) – system and software quality models

Kitchenham B, Charters S (2007) Guidelines for performing systematic literature reviews in software engineering. In: Technical report, Ver. 2.3 EBSE Technical Report. EBSE, Keele University and Durham University Joint Report, URL

`http://www.dur.ac.uk/ebse/resources/Systematic-reviews-5-8.pdf`

Kitchenham B, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S (2009) Systematic literature reviews in software engineering–a systematic literature review. Information and Software Technology 51(1):7–15, DOI 10.1016/j.infsof.2008.09.009

Kitchenham B, Pretorius R, Budgen D, Brereton OP, Turner M, Niazi M, Linkman S (2010) Systematic literature reviews in software engineering–a tertiary study. Information and Software Technology 52(8):792–805, DOI 10.1016/j.infsof.2010.03.006

Kitchenham BA, Dyba T, Jorgensen M (2004) Evidence-based software engineering. In: Proceedings of the 26th international conference on software engineering, IEEE Computer Society, pp 273–281, DOI 10.1109/ICSE.2004.1317449

Kitchenham BA, Budgen D, Brereton OP (2011) Using mapping studies as the basis for further research–a participant-observer case study. Information and Software Technology 53(6):638–651, DOI 10.1016/j.infsof.2010.12.011

Mohagheghi P, Conradi R (2007) Quality, productivity and economic benefits of software reuse: a review of industrial studies. Empirical Software Engineering 12(5):471–516, DOI 10.1007/s10664-007-9040-x

Mohagheghi P, Gilani W, Stefanescu A, Fernandez MA (2013) An empirical study of the state of the practice and acceptance of model-driven engineering in four industrial cases. Empirical Software Engineering 18(1):89–116, DOI 10.1007/s10664-012-9196-x

OMG (2015) OMG Unified Modeling Language ™(OMG UML). Tech. Rep. formal/2015-03-01, Object Management Group, URL `http://www.omg.org/spec/UML/2.5`

Petersen K, Feldt R, Mujtaba S, Mattsson M (2008) Systematic mapping studies in software engineering. In: 12th International Conference on Evaluation and Assessment in Software Engineering

Petre M (2013) Uml in practice. In: Proceedings of the 2013 International Conference on Software Engineering, IEEE Press, pp 722–731, DOI 10.1109/ICSE.2013.6606618

Petticrew M, Roberts H (2008) Systematic reviews in the social sciences: A practical guide. John Wiley & Sons

Schmidt DC (2006) Guest editor's introduction: Model-driven engineering. Computer 39(2):0025–31, DOI 10.1109/MC.2006.58

da Silva AR (2015) Model-driven engineering: A survey supported by the unified conceptual model. Computer Languages, Systems & Structures 43:139–155, DOI 10.1016/j.cl.2015.06.001

Torchiano M, Tomassetti F, Ricca F, Tiso A, Reggio G (2013) Relevance, benefits, and problems of software modelling and model driven techniques—a survey in the Italian industry. Journal of Systems and Software 86(8):2110–2126, DOI 10.1016/j.jss.2013.03.084

Whittle J, Hutchinson J, Rouncefield M, Burden H, Heldal R (2015) A taxonomy of tool-related issues affecting the adoption of model-driven engineering. Software & Systems Modeling pp 1–19, DOI 10.1007/s10270-015-0487-8

Wohlin C (2014) Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, ACM, New York, NY, USA, EASE '14, pp 38:1–38:10, DOI 10.1145/2601248.2601268

Zhou Y, Zhang H, Huang X, Yang S, Babar MA, Tang H (2015) Quality assessment of systematic reviews in software engineering: A tertiary study. In: Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering, ACM, New York, NY, USA, EASE '15, pp 14:1–14:14, DOI 10.1145/2745802.2745815

Included Secondary Studies

Budgen D, Burn AJ, Brereton OP, Kitchenham BA, Pretorius R (2011) Empirical evidence about the UML: a systematic literature review. Software: Practice and Experience 41(4):363–392, DOI 10.1002/spe.1009

Delgado A, Ruiz F, de Guzmán IGR, Piattini M (2013) Main principles on the integration of SOC and MDD paradigms to business processes: A systematic review. In: Software and Data Technologies, Springer, pp 88–108, DOI 10.1007/978-3-642-29578-2_6

Domínguez E, Pérez B, Rubio ÁL, Zapata MA (2012) A systematic review of code generation proposals from state machine specifications. Information and Software Technology 54(10):1045–1066, DOI 10.1016/j.infsof.2012.04.008

Genero M, Fernández-Saez AM, Nelson HJ, Poels G, Piattini M (2011) A systematic literature review on the quality of UML models. Journal of Database Management 22(3):46–70, DOI 10.4018/978-1-4666-2044-5.ch012

Giachetti G, Valverde F, Marín B (2012) Interoperability for model-driven development: Current state and future challenges. In: Sixth International Conference on Research Challenges in Information Science (RCIS 2012), IEEE, pp 1–10, DOI 10.1109/RCIS.2012.6240445

Giraldo FD, España S, Pastor O (2014) Analysing the concept of quality in model-driven engineering literature: A systematic review. In: Eighth International Conference on Research Challenges in Information Science (RCIS 2014), IEEE, pp 1–12, DOI 10.1109/RCIS.2014.6861030

Giraldo FD, España S, Giraldo WJ, Pastor O (2015) Modelling language quality evaluation in model-driven information systems engineering: a roadmap. In: 9th International Conference on Research Challenges in Information Science (RCIS 2015), IEEE, pp 64–69, DOI 10.1109/RCIS.2015.7128864

González CA, Cabot J (2014) Formal verification of static software models in mde: A systematic review. Information and Software Technology 56(8):821–838, DOI 10.1016/j.infsof.2014.03.003

Hansson S, Zhao Y, Burden H (2014) How MAD are we? empirical evidence for model-driven agile development. In: 3rd Workshop on Extreme Modeling, XM 2014, CEUR Workshop Proceedings, vol 1239, pp 2–11, URL http://ceur-ws.org/Vol-1239/XM2014.pdf#page=9

Jensen J, Jaatun MG (2011) Security in model driven development: A survey. In: Availability, Reliability and Security (ARES), 2011 Sixth International Conference on, IEEE, pp 704–709, DOI 10.1109/ARES.2011.110

Loniewski G, Insfran E, Abrahão S (2010) A systematic review of the use of requirements engineering techniques in model-driven development. In: Model driven engineering languages and systems, Springer, pp 213–227, DOI 10.1007/978-3-642-16129-2_16

Lucas FJ, Molina F, Toval A (2009) A systematic review of UML model consistency management. Information and Software Technology 51(12):1631–1645, DOI 10.1016/j.infsof.2009.04.009

Malavolta I, Muccini H (2014) A study on MDE approaches for engineering wireless sensor networks. In: Software Engineering and Advanced Applications (SEAA), 2014 40th EUROMICRO Conference on, IEEE, pp 149–157, DOI 10.1109/SEAA.2014.61

Mehmood A, Jawawi DN (2013) Aspect-oriented model-driven code generation: A systematic mapping study. Information and Software Technology 55(2):395–411, DOI 10.1016/j.infsof.2012.09.003

Misbhauddin M, Alshayeb M (2015) Uml model refactoring: a systematic literature review. Empirical Software Engineering 20(1):206–251, DOI 10.1007/s10664-013-9283-7

Mohagheghi P, Dehlen V (2008) Where is the proof? - a review of experiences from applying mde in industry. In: Schieferdecker I, Hartman A (eds) Model Driven Architecture – Foundations and Applications, Lecture Notes in Computer Science, vol 5095, Springer Berlin Heidelberg, pp 432–443, DOI 10.1007/978-3-540-69100-6_31

Mohagheghi P, Dehlen V, Neple T (2009) Definitions and approaches to model quality in model-based software development – a review of literature. Information and Software Technology 51(12):1646 – 1669, DOI 10.1016/j.infsof.2009.04.004

Neto VVG, Guessi M, Oliveira LBR, Oquendo F, Nakagawa EY (2014) Investigating the model-driven development for systems-of-systems. In: Proceedings of the 2014 European Conference on Software Architecture Workshops, ACM, New York, NY, USA, ECSAW '14, pp 22:1–22:8, DOI 10.1145/2642803.2642825

Nguyen PH, Klein J, le Traon Y, Kramer ME (2013) A systematic review of model-driven security. In: 20th Asia-Pacific Software Engineering Conference (APSEC 2013), vol 1, pp 432–441, DOI 10.1109/APSEC.2013.64

Santiago I, Jiménez Á, Vara JM, De Castro V, Bollati VA, Marcos E (2012) Model-driven engineering as a new landscape for traceability management: A systematic literature review. Information and Software Technology 54(12):1340–1356, DOI 10.1016/j.infsof.2012.07.008

Szvetits M, Zdun U (2013) Systematic literature review of the objectives, techniques, kinds, and architectures of models at runtime. Software & Systems Modeling pp 1–39, DOI 10.1007/s10270-013-0394-9

Yue T, Briand LC, Labiche Y (2011) A systematic review of transformation approaches between user requirements and analysis models. Requirements Engineering 16(2):75–99, DOI 10.1007/s00766-010-0111-y