

MDE in the Wild: An Exploratory Analysis on What Developers are Discussing from Q&A Platforms

Carlos Alberto Medeiros, Alan Bandeira, Paulo Henrique M. Maia
{carlos.fernandes,alan.bandeira}@aluno.uece.br,pauloh.maia@uece.br
State University of Ceará
Fortaleza, Ceará

Matheus Paixao
matheus.paixao@unifor.br
University of Fortaleza
Fortaleza, Ceará

ABSTRACT

Model-Driven Engineering (MDE) is an approach that considers models as first-class entities used in the software analysis, design and implementation. Although MDE has become popular in both academia and industry as an alternative for tackling the growing complexity of modern software, MDE has not been largely adopted in the software development process. A possible way to mitigate those problems consists of understanding how MDE has been applied in practice and what are the main barriers found by developers. Hence, this paper presents an exploratory study to analyse the discussions about MDE from two Q&A platforms: Stack Overflow and Software Engineering Stack Exchange. One hundred fourteen discussions have been analysed under four perspectives: discussion type, application domain, tools, and developer interest. As a result, we identified that 69,30% of the discussions regard technical aspects, from which metamodeling is the most discussed topic, and Xtext is the most discussed tool. In addition, we observed that discussions in which developers suggest MDE-related tools tend to attract more views and answers.

CCS CONCEPTS

• Information systems → Data mining.

KEYWORDS

Model-Driven Engineering, Mining Software Repositories

ACM Reference Format:

Carlos Alberto Medeiros, Alan Bandeira, Paulo Henrique M. Maia and Matheus Paixao. 2020. MDE in the Wild: An Exploratory Analysis on What Developers are Discussing from Q&A Platforms. In *34th Brazilian Symposium on Software Engineering (SBES '20)*, October 21–23, 2020, Natal, Brazil. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3422392.3422447>

1 INTRODUCTION

Model-Driven Engineering (MDE) emerged as a field of interest in the area of software engineering for both researchers and practitioners. This paradigm revolves around the usage of software

models in order to improve productivity and quality aspects, such as software maintainability and interoperability [10]. Due to its approach to mitigate the increasing modern software complexity, MDE has become popular in the academy [16] and industry [21]. Additionally, the techniques employed by MDE not only work in favour of developing new software, but they are also well suited for re-engineering legacy systems and automating dynamic configurations in executing systems [10]. Abstraction, automation, and analysis are MDE's key aspects in dealing with the complexity and the heterogeneity of software systems. Hence, MDE's high-level and language-agnostic artefacts have been observed supporting the expression of complex domain concepts to stakeholders

MDE has been successfully adopted in different sectors of industry, such as automobile, aerospace, telecommunications and information systems [22, 43]. In spite of its success on niche applications, MDE has not yet reached widespread acceptance as a software development practice among developers [31], as opposed to household practices such as DDD, TDD, and DevOps, for instance. Researchers argue this may be due to different reasons, including the lack of MDE-related teaching in graduate and undergraduate programs [47], developers' inexperience, and lack of training [41], misinformation over concepts [43], and lack of appropriate tooling [44].

A possible way to mitigate these issues consists on understanding how MDE has been applied in practice, focusing on the identification of the main barriers found by developers, including concerns about concepts, tools and application usage. As a result, new and improved tools may be provided by tool builders and vendors to equip a growing MDE usage community, graduate and undergraduate programmes may be better suited to show the scenarios in which MDE usage is advantageous and, finally, a roadmap for the MDE's research community can be provided.

Previous work has investigated MDE usage in practice through both surveys [22, 41, 44] and data mining [24] approaches. The surveys commonly focus on current users of MDE in industry, where topics such as - *What are the benefits of using MDE in the software development process? What are the main difficulties/problems in using MDE? What types of model transformations (M2M/M2T) are used?* - are addressed. The participants are developers with proper MDE training and experience, who work on domains where MDE is an established development practice. Although these works provide relevant insights on how MDE is successfully employed in practice, they fall short in investigating the struggles and main barriers practitioners may have when learning MDE's techniques and adopting them in their own practices.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SBES '20, October 21–23, 2020, Natal, Brazil

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-8753-8/20/09...\$15.00

<https://doi.org/10.1145/3422392.3422447>

The work by Kahani et al. [24] mined data from Eclipse’s MDE-related discussion forums, where the authors identified some of the problems and issues developers have with Eclipse’s MDE tools. Despite filling part of the gap related to understanding the barriers developers face in MDE, it only focuses on the Eclipse ecosystem, which does not represent the complete MDE’s tools landscape (see Section 3.2). Furthermore, discussions on MDE-specific forums, such as the one used in the previous work, occur between developers who are somehow familiar with MDE concepts, techniques, and tools. As previously depicted, the main reasons for the lack of widespread MDE adoption, according to researchers, revolve around entrance barriers (lack of teaching, developers inexperience, and misinformation over concepts). Hence, data from MDE-specific forums may not be ideal to understand the primordial issues developers face being introduced to MDE-based development.

Thus, we turned our attention to less specialised question-and-answer (Q&A) forums, such as Stack Overflow¹ (SO) and the Software Engineering Stack Exchange² (SESE). In SO, for example, computer science practitioners seek assistance regarding both technical and conceptual questions, increasing their technical knowledge by discussing the best practices and most appropriate tools for certain problems. Answers are validated by more experienced community members, which support the building and sharing of collective knowledge. SO has become a paramount asset for contemporary software development. By going through SO, one can find discussions at different levels of abstraction and detail, from the basic understanding of concepts³ to the inner workings of specific tools and technologies⁴. In SESE, practitioners discuss a variety of topics regarding the software engineering discipline, where the focus is on high-level concepts, such as theories, best practices, and general software development knowledge. Thus, the data on SO and SESE allow us to investigate MDE practices at different levels, from practitioners trying to grasp MDE’s basic concepts to the main barriers faced by current MDE practitioners.

Therefore, in this work, we conduct an exploratory analysis of MDE-related discussions from both SO and SESE forums under four perspectives: discussion type, application domain, tools, and developer interest. To do that, we applied techniques from mining software repositories to extract and group the main MDE discussions in both forums. In the end, we analysed a curated set of 114 discussions that occurred from August 2008 to April 2020. The contributions of this paper are threefold: (i) a categorisation and analysis of MDE-related discussions in Q&A forums; (ii) a replicable framework for extracting subjects from discussions in SO and SESE; (iii) a manually curated dataset of MDE discussions. SO data has been successfully used in empirical studies regarding different domains, such as microservices [2], mobile applications [19], cloud computing [32], and software architecture [35]. However, to the best of our knowledge, there is no work that investigates MDE practices in either SO or SESE.

The remainder of this paper is divided as follows: Section 2 presents the background that supports the work. Section 3 describes the methodology we employed, while the analyses are presented in

Section 4. Section 5 discusses the main related work and Section 6 presents the main threats to the study’s validity. Finally, Section 7 draws the conclusions and points out future work directions.

2 BACKGROUND

This section describes the background of this work. It presents an overview of Model-Driven Engineering and Mining Software Repositories.

2.1 Model-Driven Engineering

Model-driven Engineering is a software engineering approach that considers software models not only for documentation purposes but rather as primary artefacts [11]. In MDE, models can be employed to automate tasks inherent to the software development process, such as analysis, design, code generation, refactoring and code translation between platforms [16]. Its major goal is to produce technologies that aid software developers to abstract from the complexities of the underlying implementation platforms [17].

Considering the model-driven context, the use of MD* abbreviations [46] are recurrently employed to represent approaches to software design, development and implementation. Examples of the MD* ecosystem of approaches are: *Model Driven Development (MDD)*, *Model Driven Architecture (MDA)*, and *Model Driven Software Engineering (MDSE)*, among others.

Models are artefacts used to capture and represent the acquired knowledge during the software process development using abstractions with a certain level of precision and detail [18]. They help to identify and specify the structure and behaviour of the system and represent the primary source for documenting, analysing, designing, constructing, deploying and maintaining a system. Another important concept in MDE is the meta-model, which specifies the abstract syntax of a modelling language, including the concepts and relationships. This abstract syntax is considered the centre of the modelling language definition and all other concerns, such as concrete syntax and semantics, are defined within those meta-models [4].

A key concept that supports model-driven engineering is the model transformation, a technique to generate a target model from a source model by performing a set of conversion rules [25]. These rules are created in terms of the elements of both meta-models (source and target) such that the resulting model conforms with its meta-model, i.e, the transformation does not produce invalid models.

There are two types of transformation: *model-to-model (M2M)* and *model-to-text (M2T)*. The first consists of converting an input model into an output model, where those models can be instances of the same or different meta-models. The latter comprises the generation of text from an input model, usually resulting in source code for a particular programming language. M2T is also commonly called *model-to-code (M2C)*.

2.2 Mining Software Repositories

The existence of extensive and rich data about software development characterises Q&A forums, such as SO and SESE, as software repositories and, therefore, the employment of Mining Software Repositories (MSR) techniques on these data sets benefits the study

¹<https://stackoverflow.com/>

²<https://softwareengineering.stackexchange.com/>

³<https://stackoverflow.com/questions/34046283>

⁴<https://stackoverflow.com/questions/37196593>

and analysis of software development concepts and best practices. Examples of such techniques consist on identifying subjects from tags used in these platforms [3, 8, 27, 49], the use of data sets composed of discussions and their historical data [1, 37], the employment of relevance and popularity metrics in order to identify community interest [33, 38], among others.

MSR is an ever-evolving discipline within the software engineering community. For the interested reader, we point to the guidelines and surveys by Thomas et al. [40], Chen et al. [9], Siddiqui and Ahmad [34] and de F. Farias et al. [12], and the MSR conference website⁵.

3 EXPERIMENTAL METHODOLOGY

The goal of this study is to investigate, analyse and understand what developers discuss regarding Model-Driven Engineering on two Q&A platforms: Stack Overflow and Software Engineering Stack Exchange. We expect that our results and observations will shine new light on how MDE is employed in practice, based on the experience of practitioners that range from novice to experienced developers. Hence, we ask the following research questions.

- **RQ1:** What MDE-related topics have developers discussed on Q&A platforms?
- **RQ2:** What are the domains in which MDE is applied?
- **RQ3:** To what extent are MDE tools discussed in Q&A platforms?
- **RQ4:** What MDE-related topics have attracted the most attention from developers?

Figure 1 depicts the methodological framework we employed in this paper. As one can see, our empirical methodology is composed of two major phases. The rest of this section details the two phases and the steps within each of them. We provide a replication package [29] containing all the data and results generated by our methodology.

3.1 Phase 1: Identifying MDE-related discussions in Q&A platforms

This phase aims at constructing a curated dataset of MDE-related discussions from both Stack Overflow and Software Engineering Stack Exchange platforms. We consider a discussion on a Q&A platform to be a combination of a question and one (or more) answers, where there is at least one answer whose author is not the author of the question [2]. According to SO's guidelines [36], comments are not considered proper answers to questions. Hence, comments were ignored in this study.

We employed the Stack Exchange Data Explorer (SEDE) tool⁶ to extract the discussions' data from both SO and SESE. SEDE provides a DBMS-like user interface in which users can submit SQL-like queries to a certain database in the Stack Exchange network. All Q&A platforms on the network have the same relational schema. Thus, we designed a unique set of queries to extract the data from both SO and SESE. All the queries we used are available on our replication package [29].

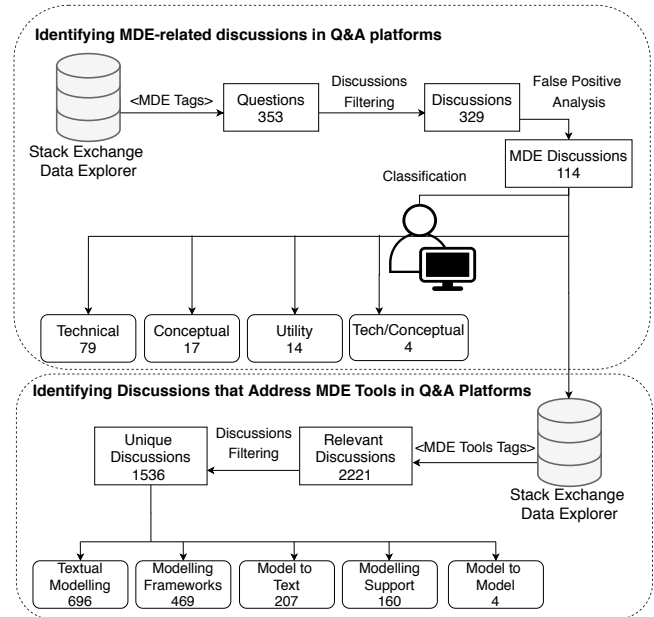


Figure 1: Empirical methodology to identify MDE-related discussions on Q&A platforms, which are further used to extract the MDE tools addressed by practitioners.

On SO and SESE, a user has the choice of selecting one or more tags to describe a question. The set of tags is mostly used to indicate the question's subjects, and to identify other users that might be able to provide an answer to the question. Hence, we used the tags to identify MDE-related discussions on both SO and SESE. To do this, we designed a procedure to assist us in determining a comprehensive set of tags practitioners use to describe MDE-related discussions.

First, based on our previous knowledge regarding MDE-based development, we started by collecting the data of all discussions on SO and SESE with the tags: "mde", "mdd", and "mda". Next, we went through all the tags that were employed in conjunction with the ones above and singled out the ones that denoted an MDE-related topic. We repeated this procedure until there were no new tags related to MDE. As a result, our final set of MDE tags is: "mde", "mdd", "mda", "mddsd", "model-driven", "model-driven-development", and "metamodel".

The set of MDE tags yielded a total of 353 questions that employed at least one of the tags. Among the questions, we filtered out the discussions according to the definition presented earlier. This resulted in 318 valid discussions obtained from SO and 11 valid discussions from SESE, summing up 329 discussions.

The next step consisted of analysing the discussions to identify false positives. i.e., discussions that employ one of the MDE tags but are not properly related to MDE. We noticed two major reasons for false-positives: (i) misusage of the tags (e.g., the question is not related to MDE but user tagged as so), and (ii) incorrect interpretation of the tag (e.g., the tag MDD referring to Microprocessor Driver

⁵<https://2020.msrfconf.org/>

⁶<https://data.stackexchange.com/>

Definition⁷ rather than Model-driven Development, while MDA referring to Managed Debugging Assistant⁸ instead of Model-driven Architecture). As a result of this analysis, we collected the final set of true MDE-related discussions, consisting of 105 discussions from SO and 9 discussions from SESE.

During a preliminary analysis of the identified MDE-related discussions, we noticed that some addressed different concerns than others. While one can observe open-ended discussions concerning basic MDE concepts⁹, one can also observe detailed discussions regarding the inner workings of specific MDE tools¹⁰. Hence, to better understand the MDE-related discussions on Q&A platforms, we classified and aggregated the discussions into categories. Previous work has done similar groupings of SO discussions by applying topic modelling approaches, such as LDA [2, 27]. However, due to the size of our dataset, we opted for a manual analysis and grouping procedure, as performed by other related work [24, 28, 42].

We employed a bottom-up manual analysis and coding strategy, where the categories emerged from the discussions. During this process, categories have been merged, refined, and curated to better represent the different types of MDE-related discussions one may find in Q&A platforms. We present the results and observations from this analysis in RQ1.

3.2 Phase 2: Identifying Discussions that Address MDE Tools in Q&A Platforms

In the second phase of our empirical methodology, we aim at identifying discussions in which practitioners address MDE tools. One should notice that a discussion that addresses an MDE tool is not necessarily MDE-related, as depicted in the previous section. Discussions where practitioners address MDE tools without employing any of the MDE-related tags are not considered MDE-related. However, to fully understand and investigate how practitioners discuss MDE's concepts, topics, and tools on Q&A platforms, we need to also collect discussions regarding MDE tools.

We consider an MDE tool to be a software tool or programming language used in at least one of MDE's common processes: meta-modelling, modelling, and M2M/M2T transformation. Note that the SESE platform was not used in this phase since tool discussion is out of its scope [13]. Therefore, only discussions from SO have been used in this phase.

To identify what MDE tools are discussed in Q&A platforms, we first need a comprehensive set of tags regarding MDE tools. Hence, we looked at all tags employed in the MDE-related discussions identified in the previous phase. Based on our previous knowledge and the work by Kahani et al. [24], we collected the first set of MDE tools tags. Next, we checked their website, git repository, and documentation to validate whether they were MDE tools or not.

At the end of this step, we identified 11 MDE tools. Each tool is presented in Table 1, alongside their correspondent tags and

Table 1: MDE tools identified in Q&A platforms. We report the tags concerning each tool and their respective purpose within MDE development

Tool	Stack Overflow Tags	Purpose
EMF	"eclipse-emf", "emf", "eclipse-emf-ecore"	Modelling Framework
Sirius	"eclipse-sirius"	
ATL	"eclipse-atl"	Model to Model
QVT	"qvt"	
Epsilon	"epsilon"	
Xtext	"xtext"	Textual Modelling
Acceleo	"acceleo"	Model to Text
Xpand	"xpand"	
Xtend	"xtend"	
OCL	"ocl"	Modelling Support
Papyrus	"papyrus"	

purpose [6]. *Eclipse Modelling Framework - EMF*¹¹, *Sirius*¹² and *Papyrus*¹³ are graphical (meta)modelling frameworks, while *Xtext*¹⁴ is used for textual modelling. The *ATL Transformation Language - ATL*¹⁵, *QVT*¹⁶ and *Epsilon*¹⁷ are M2M transformation tools, while *Acceleo*, *Xpand*¹⁸, and *Xtend*¹⁹ are M2T transformation tools. Finally, *Object Constraint Language - OCL*²⁰ is a language used to specify restrictions on models, which is considered a modelling support tool.

Based on the MDE tools tags, we further collected 2,221 discussions in which practitioner address at least one MDE tool. We discuss and analyse these observations in RQ3. All the data and results of our empirical methodology are available in our replication package [29].

4 RESULTS AND DISCUSSIONS

This section presents and discusses the answers of the research questions posed in Section 3 after mining the SO and SESE repositories following the process described in Figure 1.

4.1 RQ1: What MDE-related topics have developers discussed on Q&A platforms?

From the 114 identified MDE-related discussions, four categories of discussion have been identified: conceptual, technical, utility, and technical/conceptual. Table 2 presents a description for each category, an excerpt from a discussion in this category, and the number of discussions in the category.

¹¹<https://projects.eclipse.org/projects/modeling.emf.emf>

¹²<https://www.eclipse.org/sirius/>

¹³<https://www.eclipse.org/papyrus/>

¹⁴<https://www.eclipse.org/Xtext/>

¹⁵<https://www.eclipse.org/atl/>

¹⁶<https://projects.eclipse.org/projects/modeling.mmt.qvt-oml>

¹⁷<https://www.eclipse.org/epsilon/>

¹⁸<https://wiki.eclipse.org/Xpand>

¹⁹<https://www.eclipse.org/xtend/>

²⁰<https://projects.eclipse.org/projects/modeling.mdt.ocl>

²¹<https://stackoverflow.com/questions/11376162>

²²<https://stackoverflow.com/questions/34046283>

²³<https://stackoverflow.com/questions/7402003>

²⁴<https://stackoverflow.com/questions/4837002>

⁷<https://stackoverflow.com/questions/6756163>

⁸<https://stackoverflow.com/questions/3921661>

⁹<https://stackoverflow.com/questions/34046283>

¹⁰<https://stackoverflow.com/questions/37196593>

Table 2: Discussion categories in Q&A platforms. We provide a description of the category, followed by an excerpt of a discussion in the category, the number of discussion within the category, and the percentage of discussions in the category

Category	Description	Excerpt	#Discussions	Percentage
Technical	Related to questions regarding tools, techniques, and process used in the MDE domain	“ACCELEO: creating file in path depending on the model structure ²¹ ”	79	69.30%
Conceptual	Related to questions concerning MDE’s concepts	“Differences between MDSD and MDD ²² ”	17	14.91%
Tech/Concep	Related to both technical (tools) and conceptual issues	“What is the difference between Acceleo and Xpand? ²³ ”	4	3.51%
Utility	Related to issues involving feasibility, benefits, risks, real-use examples, and others	“I don’t like MDD but like UML - why should I use MDD if I think it is useless? ²⁴ ”	14	12.28%

Table 3: Main topics addressed in technical discussions. For each topic, we provide the number of discussions within each topic (RQ1), and the number of views and answers for the discussions within the topic (RQ4)

Topic	Discussion	View Count	Answer Count
Metamodelling	27.27%	20.23%	30.08%
Tool Suggestion	19.48%	27.77%	20.30%
M2T Transformation	18.18%	20.87%	16.54%
Constraint on Model	9.09%	5.74%	6.77%
Tools Integration and Interoperability	7.79%	9.83%	9.02%
M2M Transformation	6.49%	4.84%	6.02%
Textual Concrete Syntax	6.49%	4.25%	4.51%
Versioning Models	2.60%	1.36%	3.76%
Plugin	1.30%	0.91%	1.50%
Example of an Application	1.30%	4.19%	1.50%

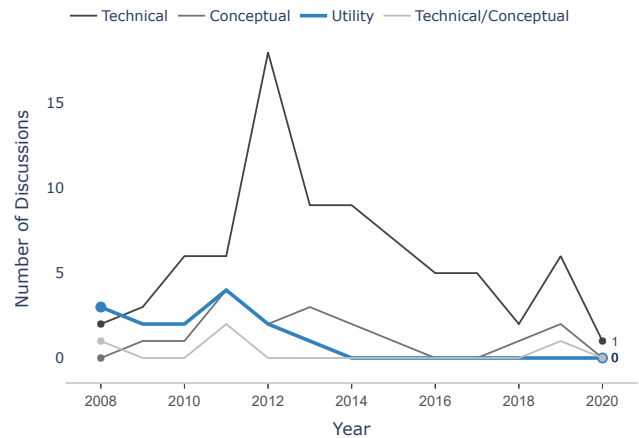
Technical discussions form the biggest category with 79 discussions, representing 69,3% of all identified discussions. These are discussions regarding tools and techniques used in the MDE domain, usually debating issues like installation, execution and recommendation. This was an expected outcome due to the nature of the consulted Q&A forums (developers usually ask technical questions on SO) and is in accordance with the results in a previous study [24].

Conceptual discussions, which address concerns regarding MDE’s foundations, came in second place enclosing 17 discussions (14,91% of the total). These discussions commonly address *what-is* and *what-differences* questions.

During the analysis of the conceptual discussions, we noticed that practitioners mixed technical and conceptual topics. For example, the question “What is the difference between Acceleo and Xpand” may look like a question regarding the difference between two tools, but the discussions actually address MDE concepts. Thus, we created the Technical/Conceptual category. As one can see, this is the smallest category, with only four discussions, and representing 3,51% of the discussions).

Finally, the last identified category involves discussions regarding general aspects of MDE, such as feasibility, benefits, risks, and others. Hence, we called it Utility, and it covers 12,28% of the discussions.

Furthermore, for each category, we performed a temporal evolution analysis of its discussions. Figure 2 illustrates the number

**Figure 2: Number of discussions per year on the Q&A platforms**

of discussions per year in the range from 2008 to 2020, which are the years of the first and last MDE-related discussions, respectively. The results indicate that technical issues have always headed the discussions among the practitioners in Q&A platforms (except from 2008), having at least one question been posted at each year, and reaching the peak of 18 questions in 2012, the triple of the number of discussions in the previous year.

Conceptual discussions, mostly concentrated from 2010 to 2015, were more present in the debates between practitioners interested in MDE’s basics. The decrease in the occurrence of conceptual discussions indicates that, gradually, the concepts of MDE have been consolidated among the community. On the other hand, the discussions that involve the utility of the MDE are concentrated only in the period from 2008 to 2013. This reinforces that the community is likely more mature regarding the role of MDE and its benefits in the software development process.

Although the discussions’ categorisation gives us a glimpse of what has been discussed in SO and SESE, it is still at a high level of abstraction. Therefore, we went through the technical discussions, which is the most representative category (nearly 70% of the posts), in order to have a more detailed view of the topics been debated.

By analysing those discussions, we identified the 10 main topics, which are depicted in Table 3. We can see “metamodelling”, one of MDE’s main activities, being the most discussed technical

Table 4: Domains Identified in Technical Discussions

Domain
Embedded/Real-time systems
Mobile Applications
Automotive industry
Web Applications
Enterprise Applications
Cloud Computing

topic, which has been addressed by 27,27% of the discussions. This may indicate that developers still face difficulties with the main (meta)modelling frameworks and tools. After that, “tool suggestion” is the second most discussed technical topic (19,48%), which shows that developers are used to asking for recommendations from their peers.

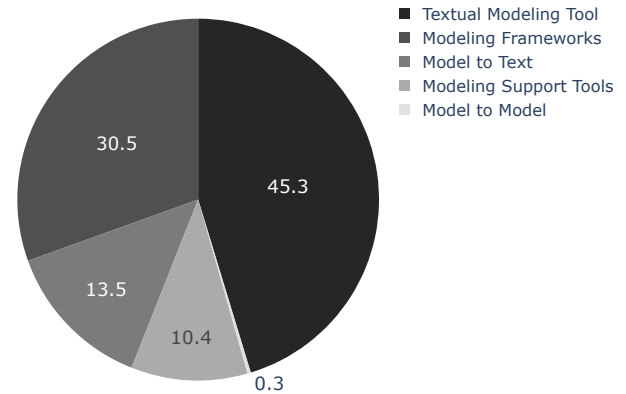
Discussions about the two MDE transformation processes, M2T and M2M, appear in third and sixth places, covering 18,18% and 6,49% of the discussions, respectively. This may point out that M2T tools attract more attention from the developers, indicating that generating code from models is a more common practice (thus incites more questions) than transforming models into other models. The other identified topics are “constraints on models” (9,09%), “integration or interoperability between tools” (7,79%), “textual concrete syntax” (6,49%), “model versioning” (2,60%), “plugins” (1,30%), and “example of application” (1,30%).

4.2 RQ2: What are the domains in which MDE is applied?

The second research question aimed at identifying the domains for which the practitioners in Q&A platforms are applying MDE. It is necessary to emphasise that we only registered the domains that the users mentioned in the discussion. In fact, the domain was in only 9 discussions. On the one hand, this avoids misinterpretation of the data, since many times the domain is not easy to determine from the discussion or it is not stated at all. On the other hand, due to being conservative, we have possibly missed different domains.

Table 4 shows the domains we could extract from the 114 MDE-related discussions. Even with our aforementioned restriction, we detected six domains: *Embedded/Real-time systems*, *Mobile Applications*, *Automotive industry*, *Web Applications*, *Enterprise Applications*, and *Cloud computing*. Our findings corroborate the study of Moghaghghi and Dehlen [30], which also identified that developers have discussed the use of MDE in cloud computing and mobile applications. The use of MDE in those domains are topics of interest in other academic research [7, 14, 20].

We highlight that this research question brings a new contribution in comparison with the work of Kahani et al. [24], who analyses the Eclipse platform’s forum but did not recognise the main application domains from the discussions.

**Figure 3: Categorisation of discussions about tools in Stack Overflow**

4.3 RQ3: To what extent are MDE tools discussed in Q&A platforms?

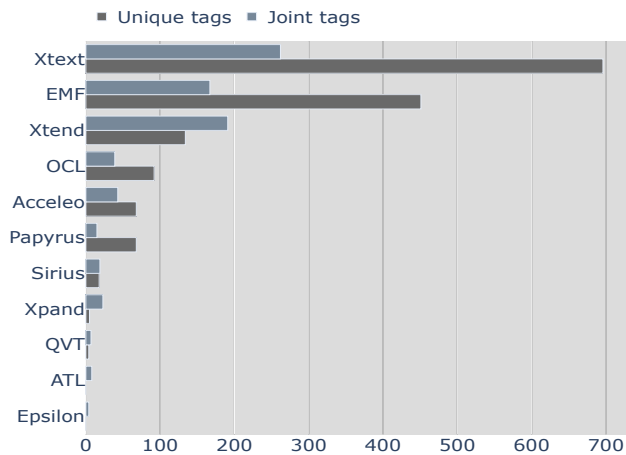
To answer this RQ, we considered the set of discussions that address MDE tools, as described in Section 3.2. Hence, we present Table 5, in which the value of cell (i,j) corresponds to the number of times a tag related to the tool i appeared in the same discussion with a tag related to the tool j . The values for cells when $i=j$ (highlighted in the table) indicate the occurrence of tags related to that tool alone. For instance, Table 5 shows that 451 discussions were tagged only with EMF-related tags, while 10 discussions used both tags related to EMF and Sirius. Note that in those 10 discussions other tools-related tags could also have been used, but due to space reasons, we limited this data analysis for pairs of tools.

Considering only the discussions with a single tool-tag, which sum up to 1,536 discussions (nearly 70% of the total), Figure 3 shows the percentage of the discussions grouped by the tool category (shown in Table 1). It is possible to observe that 45,3% of the selected discussions address textual modelling tools, even though we only identified the Xtext tool in this category. The second most tagged tool category was modelling frameworks, with 30,5%. Therefore, those two categories are responsible for more than 3/4 of the discussions, showing that (meta)modelling is by far the most discussed type of tool. This is in accordance with Table 3, which shows that metamodelling is the most discussed technical topic. This may be due to the fact that metamodelling is commonly the starting point to MDE’s ecosystem since the other types of tools (M2M, M2T and support tools) demand that an origin model exists. On the other hand, less than 1% of the posts discussed M2M tools. This shows that M2M is more common in academic studies than in real-world practices.

Taking into account the results shown in Table 5, it is possible to analyse the most discussed tools for each MDE activity (Section 2.1). Regarding metamodelling, the *EMF* tool presents the highest relevance. This shows that developers are using metamodels as a source for other tools to generate software artefacts. Besides, *EMF* is considered a standard that promotes the development of interoperable

Table 5: Number of discussions in Q&A platforms addressing different MDE tools

Category	Modelling Frameworks		Model to Model			Textual Modelling Tool	Model to Text			Modelling Support Tools	
	EMF	Sirius	ATL	QVT	Epsilon	Xtext	Acceleo	Xpand	Xtend	OCL	Papyrus
EMF	451	10	5	6	2	73	22	3	18	24	4
Sirius	10	18	0	0	0	5	1	0	2	1	0
ATL	5	0	0	0	1	2	0	0	0	0	0
QVT	6	0	0	4	0	0	0	0	0	0	1
Epsilon	2	0	1	0	0	0	0	0	0	1	0
Xtext	73	5	2	0	0	696	3	13	164	1	1
Acceleo	22	1	0	0	0	3	68	1	1	9	6
Xpand	3	0	0	0	0	13	1	5	6	0	0
Xtend	18	2	0	0	0	164	1	6	134	0	0
OCL	24	1	0	0	1	1	9	0	0	92	3
Papyrus	4	0	0	1	0	1	6	0	0	3	68

**Figure 4: Use of unique and joint tags for MDE tool-related discussions**

tools [23], corroborating the fact that it was used in all categories (Table 5). Hence, we advocate that *EMF* is a strong candidate to be taught in MDE courses in graduate and undergraduate levels.

In the context of tools that perform transformations between models (M2M), the *QVT* tool has generated more discussions in the community. On the other hand, *Xtext* is the only textual modelling tool discussed. This tool can be used for creating concrete syntax for the development of domain-specific languages. Furthermore, the interest of the community in this type of tool is in line with the results obtained by Hutchinson et al. [22], showing that many developers have used DSLs in personal projects. With respect to the M2T tools, *Xtend* appears as the most discussed tool, ahead of *Acceleo*, which is considered the protagonist of this kind of tool [6].

In addition, it is interesting to note that, when considering joint tags, the pairs *EMF + Xtext* and *Xtext + Xtend*, with 73 and 164 discussions, respectively, are dominant. This may indicate that developers are comparing graphical and textual concrete syntax using *EMF* and *Xtext*, respectively, and using *Xtend* to generate

code from a DSL created with *Xtext*. After that, we highlight the pair *EMF + OCL* (24 discussions), *EMF + Xtend* (18 discussions), *Xtext + Xpand* (13 discussions), *EMF + Sirius* (10 discussions) and *OCL + Acceleo* (9 discussions).

Another fact that draws our attention is that some tools are not discussed alone, i.e., there are no questions with only one of those tags (e.g. *ATL* and *Epsilon*), or were tagged with other tools more than by themselves (e.g. *QVT*, *Xpand* and *Xtend*). This can be seen in Figure 4. This may indicate that developers are having more difficulties when using those tools in conjunction with others.

Finally, in the domain of modelling support tools, the most discussed one is *OCL*, showing that model constraint is a concern about developers. Besides, it is common to find questions involving *OCL* with *EMF*, since it is natural that those restrictions are applied at the metamodel level.

Making a comparison with the work of Kahani et al. [24], tools related to metamodeling are more discussed on the Eclipse forums. Next, textual modelling tools (*Xtext*) are among the tools that generate the most discussions in the Eclipse forums. On the other hand, in the context of Stack Overflow, we showed that textual modelling tools are the most discussed in the community, while metamodeling tools come after.

4.4 RQ4: What MDE-related topics have attracted the most attention from developers?

This RQ aims at identifying the most “popular” discussions. To do that, we relied on two metrics provided by Stack Overflow: *ViewCount* and *AnswerCount*. Similar analyses have already been done by related studies that investigate data from SO [2, 5].

Table 6 depicts the total number of views and answers for each discussion category. One can see that Technical discussions is the most viewed and answered category, followed by the Utility category. Conceptual and Technical/Conceptual are the third and fourth categories. Comparing with Table 2, we see that, although the Conceptual category has more discussions than Utility, their discussions are less viewed and answered than its counterpart. This shows that people are more interested and interact when discussions address

Table 6: Metrics per categories

	Technical	Conceptual	Utility	Tech/ Concep
ViewCnt	77157	17170	25597	3092
AnswerCnt	137	47	67	13

problems and issues regarding the application of MDE in a real setting. This is also in tandem with the goals of Q&A forums. Furthermore, the same category was the most viewed and answered, addressing reasons for why MDE is not being widely used in the software development process. This is still an open question that draws the community's interest.

In a more detailed perspective, Tables 7 and 8 show the ranks of the top 10 viewed and answered discussions in our entire dataset, respectively. Note that the most viewed and answered discussion belongs to the Utility category, even though this category is in second place considering the overall metrics. In addition, it is interesting to see in Table 8 that only four technical discussions are among the 10 most answered and none of them is in the top 4. This indicates that, for MDE, users prefer to answer Utility discussions than exchange technical concerns.

When analysing the technical topics, presented in Table 3, we can observe that *metamodelling* was the most discussed and answered topic. However, it was ranked as 3rd considering the number of views. On the other hand, *tool suggestion*, the second most discussed topic, is the most viewed one. A possible explanation is that discussions regarding *tool suggestion* already cover the main issues developers face.

5 RELATED WORK

Villanes et al. [45] present an empirical study that aims at identifying the practices related to mobile application testing, specifically on the Android platform. The authors analysed approximately 18,000 discussions, through September 2008 to March 2017, extracted from Stack Overflow. Similarly, but focusing on software testing in general, Kochhar [26] also mined the Stack Overflow repository, collecting over than 38,000 questions from January 2009 to December 2014. Rahman [32] performed data mining on Stack Overflow to understand patterns and unknown effects of distributed denial of service (DDoS) using discussions related to the subject. The data collected from the analysis served as input for proposing a novel framework focused on protecting the cloud structure provided by AWS, against DDoS attacks. Although the aforementioned work proposes the use of Q&A forums to analyse a particular aspect or domain of the software development process, none of them focused on understanding how developers are discussing MDE topics.

The closest work to ours is presented by Kahani et al. [24], which aims at figuring out the primary questions responsible for guiding developers on the use of MDE and the ones commonly faced by newcomers of the MDE community. To do that, the researchers conduct a qualitative and quantitative survey with the questions found on the Eclipse's discussion forum, with data ranging from 2002 to 2015. The authors conclude that the subjects that draw the majority of questions are related to configuration and interoperability. In addition, when considering new users, most of them (59%) request

help on code level problems rather than conceptual questions. The authors also mention problems related to plugins and lack of documentation. However, by focusing only on the Eclipse's forums, the authors restricted the research to one technology-specific platform, which limits the results and may not capture problems faced by developers considering other tools outside the Eclipse ecosystem. Moreover, our work investigates other aspects of MDE discussions, such as popularity.

6 THREATS TO VALIDITY

In this section are discussed the threats to validity of the experiment results with regards to: (i) *Conclusion Validity*; (ii) *External Validity*; (iii) *Internal Validity*; and (iv) *Construct Validity*.

Conclusion Validity refers to the relationship between the treatment and the outcome [48]. Based on this, all the steps presented in this research involved the participation of four researchers: two master students and two Ph.D. professors in Software Engineering. Periodic meetings to discuss and improve the proposed methodology were held in order to minimise this threat. In addition, it is important to mention that the same methodology was used to extract information from the two repositories: Stack Overflow and Software Engineering Stack Exchange.

External Validity are the conditions that limit the ability to generalise the results of an experiment [48]. In this case, the analyses presented in this work may not represent the entire MDE developer community, given that only data from two Q&A platforms were investigated. However, it is out of the scope of this work to give the perception of all MDE users. In fact, we decided to use general Q&A platforms in order to have a broader cover of MDE discussions. **Internal Validity** refers to the dimension in which the results of the study can be attributed to the treatments used in the study [39]. In other words, the researcher must try to control the factors or variables that may influence the results. In the context of this work, the results obtained through data analysis could suffer interference and cause a bias generated by the authors. To minimise the impact of this threat, a rigorous methodology was developed based on knowledge of the literature on the subject under study and on information extracted from both Q&A platforms.

Construct Validity concern generalising the experiment results to the theory behind the experiment [48]. To minimise this threat, the theoretical basis was adopted following methods present in other studies of the literature and in the use of popularity metrics provided by the platforms to evaluate a discussion based on different perspectives.

Overall, we make all our data and results available in a replication package to allow for full reproducibility of the work [29].

7 CONCLUSION

This paper presented results derived from the empirical analysis of two Q&A platforms, Stack Overflow and Software Engineering Stack Exchange, to understand what practitioners discuss regarding Model-driven Engineering. To do that, we followed a two-phase methodology to identify (i) MDE-related discussions, and (ii) discussions that address MDE tools.

From the analysed data, it is possible to observe that the community shows high interest in metamodelling tools. This result

Table 7: Discussions with most views

Rank	Main Subject	Category	Quantity	Source
1	Problems with MDD	Utility	13369	https://softwareengineering.stackexchange.com/questions/55679
2	Tools Suggestion	Technical	7960	https://stackoverflow.com/questions/9775717
3	Tools Suggestion	Technical	7722	https://stackoverflow.com/questions/620188
4	UML	Conceptual	6133	https://stackoverflow.com/questions/2485746
5	M2T	Technical	4527	https://stackoverflow.com/questions/12458852
6	Integration/Interoperability between tools	Technical	4274	https://stackoverflow.com/questions/23413096
7	M2T	Technical	3867	https://stackoverflow.com/questions/8202231
8	Example of an Application	Technical	3210	https://stackoverflow.com/questions/4479883
9	Metamodelling	Technical	3096	https://stackoverflow.com/questions/4149579
10	Constraints on Model	Technical	2941	https://stackoverflow.com/questions/19705929

Table 8: Discussions with most answers

Rank	Main Subject	Category	Quantity	Source
1	Problems with MDD	Utility	17	https://softwareengineering.stackexchange.com/questions/55679
2	UML	Conceptual	12	https://stackoverflow.com/questions/2485746
3	The future of MD*	Utility	9	https://stackoverflow.com/questions/21091
4	Benefits of MDD	Utility	8	https://stackoverflow.com/questions/70781
5	Metamodelling	Technical	7	https://stackoverflow.com/questions/51660
6	Integration/Interoperability between tools	Technical	6	https://stackoverflow.com/questions/4109062
7	Metamodelling	Technical	6	https://stackoverflow.com/questions/2652918
8	Tools Suggestion	Technical	6	https://stackoverflow.com/questions/620188
9	Differences between M2T tools	Tech/Conc	5	https://stackoverflow.com/questions/7402003
10	Tools/Best practices for MDD	Tech/Conc	5	https://stackoverflow.com/questions/262279

contrasts with previous studies on the literature which point that only a small group of developers use metamodelling and code generation languages [15]. Furthermore, developers search for more information related to M2T transformations instead of M2M transformations. In addition, the community shows interest in applying restrictions to their models and integrating tools. As a result, questions involving suggestions and recommendations of MDE tools often attract practitioners' attention.

Employing MDE on software development requires direct training on MDE specific tools [22]. Hence, there is a need for an additional effort to employ MDE tools and techniques, as well as the lack of technical support on these tool-sets [44]. Therefore, discussions on the analysed Q&A forums tend to draw attention when addressing the comprehension and understanding of MDE-related concepts.

Finally, Tables 2 and 3 shed light into the question raised in Section 1 about identifying the main barriers faced by current MDE practitioners. Table 2 shows that technical barriers are the ones most faced by practitioners, where 69.3% of the discussions belong to this category. Table 3 shows that metamodelling, tools usage, and model to text transformation are the technical barriers practitioners face the most.

As future work, we intend to explore new data sources, such as Reddit, Quora, and Twitter, related to MDE to broaden and add new

perspectives in discussions regarding this subject, including more qualitative analyses of the research questions.

ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

REFERENCES

- [1] Sebastian Baltes, Lorik Dumani, Christoph Treude, and Stephan Diehl. 2018. SOTorrent: Reconstructing and Analyzing the Evolution of Stack Overflow Posts. In *Proceedings of the 15th International Conference on Mining Software Repositories (Gothenburg, Sweden) (MSR '18)*. ACM, New York, NY, USA, 319–330. <https://doi.org/10.1145/3196398.3196430>
- [2] Alan Bandeira, Carlos Alberto Medeiros, Matheus Paixao, and Paulo Henrique Maia. 2019. We Need to Talk About Microservices: An Analysis from the Discussions on StackOverflow. In *Proceedings of the 16th International Conference on Mining Software Repositories (Montreal, Quebec, Canada) (MSR '19)*. IEEE Press, Piscataway, NJ, USA, 255–259. <https://doi.org/10.1109/MSR.2019.00051>
- [3] Anton Barua, Stephen W Thomas, and Ahmed E Hassan. 2014. What are developers talking about? an analysis of topics and trends in stack overflow. *Empirical Software Engineering* 19, 3 (2014), 619–654.
- [4] S. Biffl, E. Maetzler, M. Wimmer, A. Lueder, and N. Schmidt. 2015. Linking and versioning support for AutomationML: A model-driven engineering perspective. In *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*. 499–506. <https://doi.org/10.1109/INDIN.2015.7281784>
- [5] A. Bosu, C. S. Corley, D. Heaton, D. Chatterji, J. C. Carver, and N. A. Kraft. 2013. Building reputation in StackOverflow: An empirical investigation. In *2013 10th Working Conference on Mining Software Repositories (MSR)*. 89–92.
- [6] Marco Brambilla, Jordi Cabot, and Manuel Wimmer. 2012. Model-driven software engineering in practice. *Synthesis Lectures on Software Engineering* 1, 1 (2012), 1–182.

- [7] Hugo Bruneliere, Jordi Cabot, and Frédéric Jouault. 2010. Combining Model-Driven Engineering and Cloud Computing. In *Modeling, Design, and Analysis for the Service Cloud - MDA4ServiceCloud'10: Workshop's 4th edition (co-located with the 6th European Conference on Modelling Foundations and Applications - ECMFA 2010)*. Paris, France. <https://hal.archives-ouvertes.fr/hal-00539168>
- [8] Hui Chen, John Google, and Kostadin Damevski. 2019. Modeling stack overflow tags and topics as a hierarchy of concepts. *Journal of Systems and Software* 156 (2019), 283–299.
- [9] Tse-Hsun Chen, Stephen W Thomas, and Ahmed E Hassan. 2016. A survey on the use of topic models when mining software repositories. *Empirical Software Engineering* 21, 5 (2016), 1843–1919.
- [10] Jesús Sánchez Cuadrado, Javier Luis Cánovas Izquierdo, and Jesús García Molina. 2014. Applying model-driven engineering in small software enterprises. *Science of Computer Programming* 89 (2014), 176–198. <https://doi.org/10.1016/j.scico.2013.04.007> Special issue on Success Stories in Model Driven Engineering.
- [11] Alberto Rodrigues da Silva. 2015. Model-driven engineering: A survey supported by the unified conceptual model. *Computer Languages, Systems & Structures* 43 (2015), 139–155. <https://doi.org/10.1016/j.cl.2015.06.001>
- [12] Mário André de F. Farias, Renato Novais, Methanias Colaço Júnior, Luís Paulo da Silva Carvalho, Manoel Mendonça, and Rodrigo Oliveira Spínola. 2016. A systematic mapping study on mining software repositories. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*. 1472–1479.
- [13] Software Engineering. 2019. *Welcome to Software Engineering Stack Exchange*. <https://softwareengineering.stackexchange.com/tour>
- [14] N. Ferry, H. Song, A. Rossini, F. Chauvel, and A. Solberg. 2014. CloudMF: Applying MDE to Tame the Complexity of Managing Multi-cloud Applications. In *2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*. 269–277. <https://doi.org/10.1109/UCC.2014.36>
- [15] Andrew Forward and Timothy C. Lethbridge. 2008. Problems and Opportunities for Model-centric Versus Code-centric Software Development: A Survey of Software Professionals. In *Proceedings of the 2008 International Workshop on Models in Software Engineering (Leipzig, Germany) (MiSE '08)*. ACM, New York, NY, USA, 27–32. <https://doi.org/10.1145/1370731.1370738>
- [16] R. France and B. Rumpe. 2007. Model-driven Development of Complex Software: A Research Roadmap. In *Future of Software Engineering (FOSE '07)*. 37–54. <https://doi.org/10.1109/FOSE.2007.14>
- [17] F. Freitas and P. H. M. Maia. 2016. JustModeling: An MDE Approach to Develop Android Business Applications. In *2016 VI Brazilian Symposium on Computing Systems Engineering (SBESC)*. 48–55.
- [18] Hassan Gomaa. 2011. *Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures* (1st ed.). Cambridge University Press, New York, NY, USA.
- [19] L. Guerrouj, S. Azad, and P. C. Rigby. 2015. The influence of App churn on App success and StackOverflow discussions. In *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*. 321–330. <https://doi.org/10.1109/SANER.2015.7081842>
- [20] Henning Heitkotter, Tim A. Majchrzak, and Herbert Kuchen. 2013. Cross-platform Model-driven Development of Mobile Applications with Md2. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing (Coimbra, Portugal) (SAC '13)*. ACM, New York, NY, USA, 526–533. <https://doi.org/10.1145/2480362.2480464>
- [21] John Hutchinson, Jon Whittle, and Mark Rouncefield. 2014. Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure. *Science of Computer Programming* 89 (2014), 144–161. <https://doi.org/10.1016/j.scico.2013.03.017> Special issue on Success Stories in Model Driven Engineering.
- [22] John Hutchinson, Jon Whittle, Mark Rouncefield, and Steinar Kristoffersen. 2011. Empirical Assessment of MDE in Industry. In *Proceedings of the 33rd International Conference on Software Engineering (Waikiki, Honolulu, HI, USA) (ICSE '11)*. ACM, New York, NY, USA, 471–480. <https://doi.org/10.1145/1985793.1985858>
- [23] Nafiseh Kahani, Mojtaba Bagherzadeh, James R Cordy, Juergen Dingel, and Daniel Varró. 2019. Survey and classification of model transformation tools. *Software & Systems Modeling* 18, 4 (2019), 2361–2397.
- [24] Nafiseh Kahani, Mojtaba Bagherzadeh, Juergen Dingel, and James R. Cordy. 2016. The Problems with Eclipse Modeling Tools: A Topic Analysis of Eclipse Forums. In *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems (Saint-malo, France) (MODELS '16)*. ACM, New York, NY, USA, 227–237. <https://doi.org/10.1145/2976767.2976773>
- [25] Anneke G. Kleppe, Jos Warmer, and Wim Bast. 2003. *MDA Explained: The Model Driven Architecture: Practice and Promise*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [26] Pavneet Singh Kochhar. 2016. Mining Testing Questions on Stack Overflow. In *Proceedings of the 5th International Workshop on Software Mining (Singapore, Singapore) (SoftwareMining 2016)*. ACM, New York, NY, USA, 32–38. <https://doi.org/10.1145/2975961.2975966>
- [27] Mario Linares-Vásquez, Bogdan Dit, and Denys Poshyvanyk. 2013. An exploratory analysis of mobile development issues using stack overflow. In *2013 10th Working Conference on Mining Software Repositories (MSR)*. IEEE, 93–96.
- [28] Tamara Lopez, Thein Tun, Arosha Bandara, Mark Levine, Bashar Nuseibeh, and Helen Sharp. 2019. An Anatomy of Security Conversations in Stack Overflow. In *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society (Montreal, Quebec, Canada) (ICSE-SEIS '19)*. IEEE Press, Piscataway, NJ, USA, 31–40. <https://doi.org/10.1109/ICSE-SEIS.2019.00012>
- [29] Carlos Alberto Medeiros, Alan Bandeira, Paulo Henrique Maia, and Matheus Paixao. 2020. Replication package for the paper: “MDE in the Wild: An Exploratory Analysis of What Developers Discuss in Q&A Platforms”. <https://zenodo.org/record/3998518>
- [30] Parastoo Mohagheghi and Vegard Dehlen. 2008. Where Is the Proof? - A Review of Experiences from Applying MDE in Industry. In *Model Driven Architecture - Foundations and Applications*, Ina Schieferdecker and Alan Hartman (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 432–443.
- [31] Gunter Mussbacher, Daniel Amyot, Ruth Breu, Jean-Michel Bruel, Betty H. C. Cheng, Philippe Collet, Benoit Combemale, Robert B. France, Rogardt Heldal, James Hill, Jörg Kienzle, Matthias Schöttle, Friedrich Steimann, Dave Stikkolorum, editor="Dingel Juergen Whittle, Jon", Wolfram Schulte, Isidro Ramos, Silvia Abrahão, and Emilio Insfran. 2014. The Wolfing of Model-Driven Engineering Thirty Years from Now. In *Model-Driven Engineering Languages and Systems*. Springer International Publishing, Cham, 183–200.
- [32] M. Rahman. 2017. Mitigating Information Disclosure Attacks in the Cloud by Blocking Invalid User and Figure Out Problems to Solve DDoS by Analyzing StackOverflow Questions. In *2017 2nd International Conference on Electrical Electronic Engineering (ICEEE)*, 1–4. <https://doi.org/10.1109/ICEEE.2017.8412901>
- [33] Leonardo Jiménez Rodríguez, Xiaoran Wang, and Jilong Kuang. 2018. Insights on Apache Spark Usage by Mining Stack Overflow Questions. In *2018 IEEE International Congress on Big Data (BigData Congress)*. IEEE, 219–223.
- [34] Tamanna Siddiqui and Ausaf Ahmad. 2018. Data mining tools and techniques for mining software repositories: a systematic review. In *Big Data Analytics*. Springer, 717–726.
- [35] M. Soliman, M. Galster, A. R. Salama, and M. Riebisch. 2016. Architectural Knowledge for Technology Decisions in Developer Communities: An Exploratory Study with StackOverflow. In *2016 13th Working IEEE/IFIP Conference on Software Architecture (WICSA)*. 128–133. <https://doi.org/10.1109/WICSA.2016.13>
- [36] StackOverflow. 2020. Stack Overflow Annual Developer Survey. <https://stackoverflow.com/help/privileges> Accessed: 2020-06-14.
- [37] Clayton Stanley and Michael D Byrne. 2013. Predicting tags for stackoverflow posts. In *Proceedings of ICCM, Vol. 2013*.
- [38] Samarth Tumbad, Rohit Nandwani, and Suzanne K McIntosh. 2020. Analyzing programming languages by community characteristics on Github and StackOverflow. *arXiv preprint arXiv:2006.01351* (2020).
- [39] Jerry R Thomas, Jack K Nelson, and Stephen J Silverman. 2009. *Métodos de pesquisa em atividade física*. Artmed Editora.
- [40] Stephen W. Thomas, Ahmed E. Hassan, and Dorothea Blostein. 2014. Mining Unstructured Software Repositories. In *Evolving Software Systems*, Tom Mens, Alexander Serebrenik, and Anthony Cleve (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 139–162.
- [41] F. Tomassetti, M. Torchiano, A. Tiso, F. Ricca, and G. Reggio. 2012. Maturity of software modelling and model driven engineering: A survey in the Italian industry. In *16th International Conference on Evaluation Assessment in Software Engineering (EASE 2012)*. 91–100. <https://doi.org/10.1049/ic.2012.0012>
- [42] C. Treude, O. Barzilay, and M. Storey. 2011. How do programmers ask and answer questions on the web?: NIER track. In *2011 33rd International Conference on Software Engineering (ICSE)*. 804–807. <https://doi.org/10.1145/1985793.1985907>
- [43] Antonio Vallecillo. 2015. On the industrial adoption of model driven engineering. Is your company ready for MDE? *International Journal of Information Systems and Software Engineering for Big Companies (IJISEBC)* 1, 1 (2015), 52–68.
- [44] A. Vetro, W. Bohm, and M. Torchiano. 2015. On the Benefits and Barriers When Adopting Software Modelling and Model Driven Techniques - An External, Differentiated Replication. In *2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. 1–4. <https://doi.org/10.1109/ESEM.2015.7321210>
- [45] Isabel K. Villanes, Silvia M. Ascate, Josias Gomes, and Arilo Claudio Dias-Neto. 2017. What Are Software Engineers Asking About Android Testing on Stack Overflow?. In *Proceedings of the 31st Brazilian Symposium on Software Engineering (Fortaleza, CE, Brazil) (SBES'17)*. ACM, New York, NY, USA, 104–113. <https://doi.org/10.1145/3131151.3131157>
- [46] Markus Volter. 2009. MD? Best practices. *Journal of Object Technology* (2009), 8(6):79–102.
- [47] J. Whittle, J. Hutchinson, and M. Rouncefield. 2014. The State of Practice in Model-Driven Engineering. *IEEE Software* 31, 3 (May 2014), 79–85. <https://doi.org/10.1109/MS.2013.65>
- [48] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. 2012. *Experimentation in software engineering*. Springer Science & Business Media.
- [49] Xin-Li Yang, David Lo, Xin Xia, Zhi-Yuan Wan, and Jian-Ling Sun. 2016. What security questions do developers ask? a large-scale study of stack overflow posts. *Journal of Computer Science and Technology* 31, 5 (2016), 910–924.