

# The Problems with Eclipse Modeling Tools: A Topic Analysis of Eclipse Forums

Nafiseh Kahani, Mojtaba Bagherzadeh, Juergen Dingel, James R. Cordy  
School of Computing, Queen's University  
Kingston, Canada  
{kahani,mojtaba,dingel,cordy}@cs.queensu.ca

## ABSTRACT

Eclipse offers a wide range of tools supporting various aspects of modeling and Model-Driven Engineering (MDE). Arguably, the Eclipse ecosystem has been and continues to be one of the most important modeling tool repositories and sources of information about these tools, with, for example, more than 180,000 posts in the modeling forums since 2002. In this paper, we collect and analyze the content of the 30 most widely used Eclipse forums associated with different modeling and MDE tools, such as EMF, Xtext, ATL, Epsilon, and GMF. Using state-of-the-art text mining techniques coupled with manual analysis, we explore these forums with respect to two important questions: What are the primary issues, problems, and challenges raised in the use of these tools? And, perhaps even more important: Which of these issues are most commonly faced by “newbies” in the MDE community? Our study provides supporting evidence for some commonly held but unproven beliefs, such that plug-ins and documentation issues are the most common, and suggests which issues actually present the biggest “barriers to entry” for new users of MDE tools, and how they might be addressed.

## Keywords

Model-Driven Engineering (MDE); Eclipse; Topic Analysis

## 1. INTRODUCTION

Deservedly or not, tools supporting Model-Driven Engineering (MDE) do not have a good reputation. They have been described as insufficient [6], complex [14], immature [21] and a barrier to adoption [24]. While most of these observations focus on technical limitations, Whittle et al. have recently suggested that the discussion about the ways MDE tools impact the adoption of MDE should be broadened [28]: Based on the results of 39 interviews in 20 different companies carried out between 2009 and 2013, they present a taxonomy of MDE tool-related issues which adds

organizational and social factors to the usual technical issues. Their analysis also taps directly into the experience of industrial developers and provides insight into what they perceive as the pros and cons of modeling and MDE.

At the same time, the Eclipse ecosystem has been providing the modeling community with a large collection of modeling tools providing support for, e.g., (1) the development of (meta-) modeling languages and associated tooling such as textual and graphical editors, (2) model transformation, (3) model development and MDE. In the context of modeling, Eclipse.org currently hosts 53 projects including Xtext, ATL, Epsilon, JET, Acceleo, Xtend, Papyrus, Graphiti, GMF, and projects supporting different modeling languages or technologies including QVT, OCL, UML2, BPMN, and EMF. Moreover, many Eclipse projects have public forums associated with them in which users can ask questions, report problems and share solutions. 39 of these forums are devoted to modeling. Since 2002, more than 180,000 posts have been made to modeling forums in an attempt to answer more than 50,000 questions. Arguably, Eclipse has become the most important source of tools and expertise for the modeling community.

In this paper, we explore the kinds of problems and barriers to adoption of MDE tools and techniques by analyzing the actual questions, issues and problems talked about in Eclipse modeling tool forums. Similarly to Whittle et al. [28], we are interested in determining which kinds of problems users have with tools and how their experience can be improved. However, we approach the problem from a different and complementary angle. Despite the relatively large number of interviews conducted, the study in [28] is essentially *qualitative* in nature, aiming to understand the impact of tools in an exploratory way with mostly manual, semi-structured or unstructured data collection methods and sample sizes small enough for manual collection and evaluation to be feasible. Our work, on the other hand, follows a more *quantitative* approach. More specifically, we extract data from the discussion forums of 30 most popular Eclipse modeling tools to shed light on the following two sets of research questions:

- RQ1: Which are the primary issues, problems, and challenges mentioned on Eclipse modeling forums and how does their frequency change over time?
- RQ2: Which issues, problems, and challenges are “newbies” (i.e., people with no or little background in the tool) facing and how many of their questions end up being answered satisfactorily?

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 ACM 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](mailto:permissions@acm.org).

MODELS '16, October 02-07, 2016, Saint-Malo, France

© 2016 ACM. ISBN 978-1-4503-4321-3/16/10...\$15.00

DOI: <http://dx.doi.org/10.1145/2976767.2976773>

**Table 1: Summary of tool categories.**

Category	Tool Name
Modeling Frameworks and Tools (MFT)	EMF, EMF Technology (e.g., Ecore Tools, EMFatic), EMF Diff/Merge, EMF Compare
Textual Modeling Tools (Xtext)	Xtext
Model Transformation Tools (MTT)	ATL, Epsilon, Eclipse M2T (e.g., Xpand, JET, Acceleo, Xtend), MoDisco, MOFScript, QVT, VIATRA, UMLX
Modeling Development Tools (MDT)	Papyrus, eTrice, UML2 Tools, UML2, OCL, XSD, AMW, BPMN2
Graphical Modeling Frameworks and Tools (GFT)	Graphiti, GMF, EMF Parsley, ELK

To answer RQ1, we use state-of-the-art Information Retrieval (IR) techniques to cluster and make sense of unstructured data. More precisely, we use a popular probabilistic topic model called Latent Dirichlet Allocation (LDA) to extract *topics* from the posts on Eclipse forums semi-automatically [5]. Roughly, a topic is a collection of words that frequently co-occur. We use these topics to determine which kinds of issues bring people to post questions on a forum. To answer RQ2, a more manual technique is used. We collect the first questions users ask after signing up for a forum, manually determine what the question is about and whether the question ultimately is answered successfully.

Our paper contributes an indication of which problems users of these Eclipse modeling tools have. This information is useful to identify concrete, effective ways how to improve the experience of different kinds of users, including changes to the tool itself (e.g., its features, GUI), to the way it is used in isolation or in concert with other tools (e.g., formats of inputs and outputs), or to supporting artifacts (e.g., documentation, case studies). Moreover, we make the collected forum data which underlies our analysis public to allow for further analysis and research [11].

The remainder of the paper is organized as follows: Section 2 describes the data sources, and data collection and analysis techniques used. Section 3 presents the results of the analysis with respect to each of the two RQs. Sections 4 and 5 discuss threats to validity and related work. Finally, Section 6 concludes the paper and discusses future work.

## 2. APPROACH

As mentioned, our work focuses on two RQs. Due to the volume of data involved, two different approaches were chosen to address them. We will first describe aspects relevant to both of them such as where and how the data was obtained, how it was partitioned, and which terminology we will use to refer to different parts of the data. Then, details about how each analysis was performed are provided.

### 2.1 Data Sources

For this study we have chosen to focus on analyzing the text content posted in Eclipse modeling forums [7]. Each forum is an online platform where users can communicate with the community of people developing and using Eclipse modeling tools. Forums allow users to ask new questions

or answer existing questions. Users can edit both questions and responses to improve their quality. Our data set includes the questions and responses posted in a wide range of tools and frameworks, such as EMF and Xtext. Out of the 39 modeling forums on Eclipse.org/forums, we pick the ones associated with a single tool and a sufficiently high number of discussions. As a result, the forums of the tools BPEL Designer, AMP, Amalgam, Eclipse B3, and Sphinx are not considered. We obtain 30 tools (forums) in total including EMF, Xtext, ATL, Epsilon, QVT, Papyrus, OCL, Graphiti, and GMF.

### 2.2 Tool Categorization

While there is a fair bit of diversity amongst the 30 tools considered, there also are clear commonalities based on, e.g., the development purpose or activity they support. For the purpose of analysis, we categorize the tools based on the commonalities between them (Table 1). We group Eclipse Modeling Framework (EMF) and its supporting tools together into a category called MFT. The EMF forms the foundation of all Eclipse modeling tools, and provides support for tools and applications built on structured data models. This support is quite extensive and includes the creation of textual editors, code generation from models, and the use of models as metamodels, i.e., language descriptions. Support is spread out over several different tools including EMF Technology, Ecore Tools, EMF Diff/Merge, and EMF Compare.

Xtext provides support for the definition and use of languages; based on a grammar describing some language, it can be used to either automatically generate or manually create various kinds of tooling supporting that language, such as parser, editors, validators, code generators, interpreters, etc. We decided to have a separate category for Xtext, because its purpose and functionality appears rather unique amongst Eclipse modeling tools.

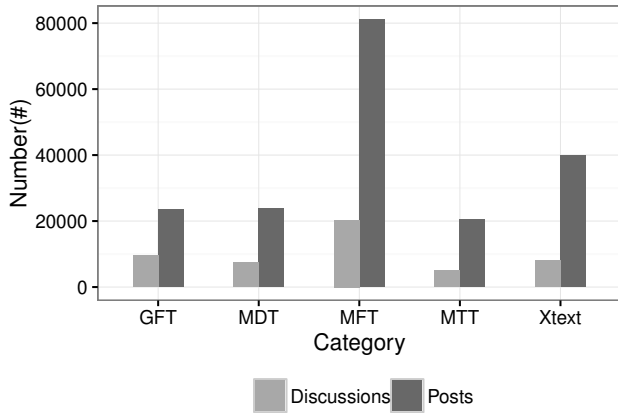
The MTT category is given by the tools devoted to model transformation. In contrast to the tools in the MFT category, the model transformation tools tend to be more “stand-alone” and independent. However, they clearly share a common purpose. In our categorization, the MTT category includes model transformation tools including ATL, Epsilon, JET, Acceleo, MOFScript, QVT, and VIATRA. Some of the tools in this category, such as Epsilon, can be used for the other purposes, such as verification. For these tools, we manually reviewed the most recent discussions in related forums, and then categorize them based on the their main discussions’ topics. Our review shows that users are mainly using Epsilon for model transformation.

Tools for the creation, manipulation, validation and use of different kinds of models form another category, called MDT. This category includes the tools Papyrus, eTrice, OCL, AMW, and BPMN2.

Another category contains tools aimed at providing support for more sophisticated forms of concrete syntax through the generation of editors for forms, tables, and graphics paired with different layout algorithms. The GFT category includes Graphiti, GMF, EMF Parsley, and ELK.

### 2.3 Data Extraction and Terminology

Since forum data is not directly available in one, single collection, we use web scraping to extract all posts from each of the forums. The following terminology is important



**Figure 1: Number of discussions and posts in each category.**

to understand the structure and semantics of the data: A *post* can either be a *question* or a *response* to a question. A question starts a *discussion* consisting of the question and all its responses. Each discussion corresponds to a *document*. There is, thus, a one-to-one correspondence between questions, discussions, and documents.

## 2.4 Approach for RQ1: Topic Models

The following approach is used by most applications of topic models to mining software repositories [5, 22]. After data extraction, the data is cleaned, analyzed automatically to extract topics, and then the topics are validated and labeled manually.

### Data Cleansing.

During the cleansing of data, we remove any code snippets. The rationale behind this step is that it has been shown that programming language syntax, keywords, and code snippets do not help topic modeling and can be considered noise [22]. Next, we remove words which do not create meaningful topics such as “is” and “the” [17], and then apply the Porter stemming algorithm [20] to map the words to their bases (also called root form).

After cleansing, our dataset includes all 50,206 questions from 30 forums containing 188,915 posts from 2002 (the beginning date of the Eclipse Modeling forums) until March 2016. The length of the questions varies between 2 and 4,236 words. Each question usually triggers between 1 and 200 responses. More than 10,000 users have participated in the discussions. Figure 1 shows the number of questions and posts in each of our five tool categories.

### Automatic Topic Extraction.

Users post their questions informally in some natural language. Due to different possible interpretations of natural language text, the effectiveness of fully automatic data mining techniques for finding topics is reduced [10]. On the other hand, it is impossible to analyze large collections of questions manually. Hence, our methodology follows standard practice and combines the use of automatic mining tools to extract topics with a manual step in which extracted topics are checked and described. The work in, e.g., [25, 27, 9],

proceeded in a similar fashion.

We use Latent Dirichlet allocation (LDA), one of the most widely used statistical topic modeling techniques, to discover the topics in Eclipse forums. Using machine learning, topic models infer latent topics that describe a corpus of text documents. A *topic* is a collection of *topic words* that co-occur in the corpus [23] with at least a certain minimal frequency. In LDA, each document is represented as a probability distribution over the discovered topics. Topic modeling examines a set of documents to find the word frequencies and co-occurrence frequencies in the documents to create a model of related words. Topic modeling has successfully been applied to a variety of domains, such as computer vision [3] and scientific topics [8], to automatically organize and analyze millions of unstructured documents. Besides, the unsupervised nature of LDA allows for automatic topic discovery which makes this technique more interesting and faster.

We use Andrew McCallum’s MALLEET tool for LDA computation [18]. The use of the tool is subject to a few user-specified parameters, the most important of which is the number of topics  $K$  to be extracted from the documents (i.e., forum discussion).  $K$  thus determines the “granularity” of the analysis and the resulting topics, with small values of  $K$  leading to fewer and more general topics, while large values of  $K$  result in more numerous and specific topics.  $K$  needs to be chosen carefully, as it influences the usefulness of the topics produced. Unfortunately, it has been shown that there is no single best choice of  $K$  for all datasets [26], meaning that different values of  $K$  may have to be tried. In our case, such experimentation led us to set  $K$  to 40. Another parameter is the number of topic words per topic. Again, after some experimentation, we set this value to 15. A Gibbs sampler is a statistical technique necessary for the implementation of LDA. MALLEET requires the number of sampling iterations to be set. As suggested in [8], we use 500 sampling iterations.

### Automatic Topic Analysis.

We use the metrics presented by Barua *et al.* [4] to rank and evaluate the importance and relationships of the extracted topics. The first metric is *topic share*, which measures the relative popularity of a topic across all discussions. I.e., a high topic share indicates that the topic appears in many discussions. The second metric is *topic trend over time*, which shows how the share of a topic changes over time. The following notation is used to define these metrics formally.

Assuming  $K$  topics  $z_1, \dots, z_K$  have been discovered by LDA, the membership of a particular topic  $z_j$  in discussion (document)  $d_i$  is denoted as  $\theta(d_i, z_j)$ . We have

$$\forall i, j : 0 \leq \theta(d_i, z_j) \leq 1 \quad \text{and} \quad \forall i : \sum_{j=1}^K \theta(d_i, z_j) = 1$$

A threshold  $\delta$  is used to keep only the main topics in each discussion and discard low frequency topics. A value of 0.10 for  $\delta$  has been shown to remove ‘noise’, while still allowing the dominant topics in each document to be identified [4]. We also set  $\delta$  to 0.10.

Equation 1 defines the *topic share* metric as the overall share of a topic  $z_j$  among all discussions in a category of

forums:

$$share(z_j) = (1/|D|) \cdot \sum_{\substack{d_i \in D \\ \theta(d_i, z_j) \geq \delta}} \theta(d_i, z_j) \quad (1)$$

where  $D$  is the set of all discussions in all forums in a particular tool category. Intuitively,  $share(z_j)$  indicates the proportion of discussions containing topic  $z_j$ .

The metric *topic trends over time* is used to analyze the temporal trends of topics [4]. For this, the *impact* of a topic  $z_j$  in year  $y$  is defined as:

$$impact(z_j, y) = (1/|D(y)|) \cdot \sum_{\substack{d_i \in D(y) \\ \theta(d_i, z_j) \geq \delta}} \theta(d_i, z_j) \quad (2)$$

where  $D(y)$  contains the set of all discussions in all forums in a particular tool category in year  $y$ . Intuitively, the  $impact(z_j, y)$  indicates the relative proportion of discussions containing topic  $z_j$  in year  $y$ .

### Manual Topic Labeling.

As mentioned, LDA extracts topics in the form of collections of topic words. Each such collection, together with some of the discussions the collection was ranked high in, needs to be inspected manually to determine its coherence and descriptiveness and to come up with a suitable *topic description*. For instance, the topic characterized by the topic words “jet templat tag xpath transform xml messag variabel eclips context express custom function emf gener” has a share of 8.50 amongst the discussions in forums of tools belonging to the MTT category; after inspection of some of the discussions that this topic appears in, a suitable description of this topic appears be “JET template issues”. Similarly, the collection of topic words “valid evl wizard constraint epsilon diagram editor ewl emf check model gmf creat view context” has a share of 8.18 amongst discussions in MTT forums and the description “EVL constraint language issues” appears suitable.

Unrelated or irrelevant topics, such as version announcements, calls for papers or participation, are excluded from our topic list. The end result of our analysis are five lists containing, one list for each of the five tool categories, the ten topics with the highest share values. Due to the page limitation, only the top seven topics will be presented in Tables 2 to 6, while the remaining three topics will be mentioned in the text accompanying the each table.

## 2.5 Approach for RQ2: Manual Classification

We first extract all newbie questions from all forums posted between January 2014 and December 2015. We choose only these two years because of time-consuming manual process. A newbie question is either (1) one of the first ten questions that a user posts on a forum, or (2) a question explicitly containing the keywords ‘beginner’, ‘newbie’, or ‘novice’. This process resulted in 2,339 questions. All of these questions were used to determine the groups of common newbie questions in Section 3.2.2 and the response rate analysis in Section 3.2.3.

For the identification of groups of newbie questions for each forum, we ignore tools with very low numbers of beginner questions. For instance, in GFT, we select Graphite and GMF because they account for 94% of the 247 questions in total; for MDT, Papyrus was selected (84% of 448

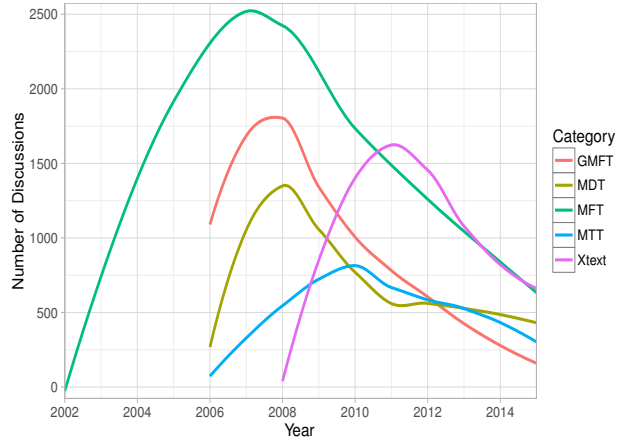


Figure 2: The number of discussions per year.

questions); and for MTT, we select Epsilon and ATL (80% of 310 questions). We then perform a manual inspection of the 1,349 newbie questions of the selected tools (Xtext: 362, MFT: 303, MDT: 296, MTT: 205, and GFT: 183) and, as much as possible, identify the issues, problems, or challenges that triggered the question. Note that a single question may contain several issues. Results are given in the latter part of Section 3.2.2.

Finally, we determine to what extent the questions have eventually been answered satisfactorily, where a satisfactory answer is one that is known to be correct, or that is eventually followed by some kind of positive acknowledgment from the person who had asked the question.

## 3. PRESENTATION OF RESULTS

We first present data showing the number discussions in each category per year from 2002 until 2015 (Figure 2). The following observations can be made: (1) All categories except MTT have experienced rather dramatic rates of increase and decrease; for MTT the initial growth till 2010 and the subsequent decrease have been a lot more gradual. (2) Discussions in MFT, GFT, and MDT peaked around 2008; MTT and Xtext peaked in 2010 and 2011, respectively. (3) MDT is the only category in which the discussion volume actually rebounded (between 2011 and 2012), albeit only by a very small amount, after the decrease following the peak; since that rebound between 2011 and 2012, the number of discussions has been the most stable of all categories.

We emphasize that declining numbers of discussions do not necessarily indicate decreasing interest in a tool. For instance, discussion numbers could also decrease because users have fewer problems (because, e.g., bugs and problems have been fixed in subsequent tool releases), or because users have fewer questions for which they cannot find an answer (e.g., on the tool forum itself or elsewhere on the Internet or literature). More data is necessary (e.g., the number of tool downloads and the tool release history) to back up explanation attempts.

The remainder of the presentation and discussion of results is partitioned into two parts, each focusing on one of the research questions mentioned in the introduction. Below, for each question, we will argue its significance, and present the analysis results for each category together with,

**Table 2: MFT Topics.**

LDA Topic Words	Description	Share	Trend
<i>element chang featur issu comment order time reason solut remov method add expect result end</i>	CRUD (Create, Read, Update, Delete) operations on models	27.94	
<i>eclips emf plugin project instal jar version depend build run plug-in bundl runtim updat workspac</i>	Plugin dependencies, installation & project creation issues	14.99	
<i>error test code bug run fix log eclips null debug fail issu open ecp version</i>	Bugs, integration with ECP	14.60	
<i>xml schema xsd element gener ecor emf document serial namespac map attribut featur root instanc</i>	XML related issues such as Ecore-to-XSD transformation using XSLT & XMI	12.69	
<i>editor edit emf view gener select open page action text tree dialog menu show displai</i>	Creating and customizing editors for EMF models	12.55	
<i>resourc load object save proxi refer resolv instanc persist emf eobject xmi uri root content</i>	Persistence API	12.49	
<i>gener interfac code annot extend method emf ecor public xcore oper featur defin inherit add</i>	Model annotation, code generation and its customization	12.31	

when appropriate, concrete suggestions for future work, either in the form of improvements to the tool and any supporting material, or in the form of future research topics.

### 3.1 Research Question 1

*Which are the primary issues, problems, and challenges mentioned on Eclipse modeling forums and how does their frequency change over time?*

#### 3.1.1 Motivation

Information about the most frequent discussion topics is potentially useful to many parties in the modeling community. On the one hand, it might inspire modeling researchers interested in developing new approaches, techniques, or tools. On the other hand, it might help forum moderators to be better prepared and the Eclipse Foundation to improve their web pages by allowing users to find relevant information more quickly via, e.g., a list of frequently asked questions (FAQ) with the most frequently occurring questions appearing in the most visible place. Also, the information would help educators to prepare students better for the use of a tool.

#### 3.1.2 Results

The extracted topics appear in Tables 2 to 6. In each table, the topics are ordered by their share. The first column in the table shows the collection of topic words produced by LDA and the second column provides a one-sentence description of the topic. The third column shows the relative share the topic as amongst all discussions in this category (see definition in Section 2.4). The fourth column (labeled ‘Trend’) contains a small graph indicating how the share of this has changed over the years.

#### Common Topics (2,339 questions from all forums).

1. *Plugin issues:* A big part of the appeal of Eclipse is its extensibility: Tools can be packaged and distributed as plugins and benefit from a powerful, rich framework made for reuse and obviating the need to implement standard functionality from scratch; tools with impressive functionality can be built quite rapidly. However, these benefits come at the price of problems due to a potentially brittle plugin architecture with intricate, insufficiently understood, or even unknown dependencies, which can cause installation problems or software failure after version upgrades. Our analy-

sis clearly reveals that this downside is significant. In fact, Tables 2-6 show that, overall, problems related to the tool’s plugin architecture, such as plugin dependencies, plugin configuration, plugin installation, and plugin (re-)deployment, are, by far, the most common source of headache for users.

For instance, plugin dependencies are the second most frequent topic in the MFT and Xtext forums, with a share of 14.99 and 18.89, respectively, affecting first-time users and senior users (as indicated by their Eclipse forum membership status) alike:

First-time user:

*“I downloaded EMF for M5 today but the EMF plugin does not get activated automatically on next Eclipse startup. I am clueless what am I missing in my install ? Is there a way to verify that the plugin was activated/not-activated. I checked the log but there were no errors.”*








Senior user:

*“The generated edit and editor plugins are not deployable to Eclipse 3.0. Is this a known problem?”*

The analysis shows that the use of Eclipse is not completely transparent and that users need to have a certain amount of knowledge of core concepts of the Eclipse framework before they can reliably install, use, and upgrade Eclipse-based tools. It appears that better Eclipse-wide support to manage plugins would be the most effective step to improve the experience of users of Eclipse (modeling) tools. The development of such support would benefit from a categorization of plugin issues and their root causes. For instance, we observed that some problems were caused by (1) attempting to generate code from the ‘GenPackage’, rather than the ‘GenModel’; or an improperly set (2) target platform, or (3) ‘model plugin ID property’.

2. *Documentation issues:* As shown in Tables 2 to 6, missing or low quality documentation, tutorials, and manuals are the second most commonly occurring problem. E.g., the following post is taken from an MTT forum:

**Table 3: Xtext Topics.**

Top LDA Words	Description	Share	Trend
<i>xtext languag dsl grammar model gener implement creat code syntax defin write document pars exampl</i>	How to create the DSL from scratch, how to convert grammar of existing language to Xtext	24.40	
<i>project plugin eclips xtext dsl creat run gener xml applic problem extens add mydsl depend</i>	Plugin dependencies, exporting Xtext projects to runnable applications or plugins	18.89	
<i>element grammar model attribut problem defin rule properti object list creat featur tabl chang</i>	Grammar rule applications	13.47	
<i>xtext eclips version instal updat error depend featur site plugin download log releas build problem</i>	Installation and update issues involving, e.g., the update site and Eclipse versions	11.55	
<i>gener code xtend dsl output method model creat compil call sourcod encod templat write igener</i>	Code generation issues, e.g., running Xtext from command line	10.23	
<i>editor xtext model open edit gmf eclips text save creat xtexteditor integr diagram content emf</i>	Customizing default Xtext editor and integrating it with existing tools	9.85	
<i>import scope refer qualifi element xtext model packag qualifiednam dsl namespac entiti grammar importuri defin</i>	Import/export, scoping, and references in grammars	9.32	

*“I need to transform a RDF/XML model to a Matlab structure. I don’t know the correct manner to create the right project type, where to locate my RDF/XML model, what type of file generate (.eol, .etl ?). I read the Epsilon Book but I’m confused about how to start.”*

However, we can also see that MFT and Xtext far better in this respect than MTT, MDT, and GFT. For MTT and GFT, documentation-related issues top the list with shares of 44.7 and 36.26, respectively, while for MDT they are in sixth place with a share of 10.7; for MFT and Xtext, the LDA analysis did not even produce a topic on documentation-related issues.

### Topics in MFT Forums (Table 2).

Posts related to ‘CRUD’ operations on models take up the largest share in this category (27.94) with a declining trend most recently. Interestingly, the share of posts suggesting more advanced uses of MFT tools has been increasing: For instance, posts related to the EMF Client Platform (ECP), a framework for building EMF-based client applications, have the third largest share (14.60). Similarly, LDA discovered a discussion topic (not shown; share: 10.63) related to Hibernate, a framework mapping object-oriented domain models to relational databases useful for, e.g., developing EMF-based business applications. Both trends might suggest increasing maturity and adoption of EMF.

Other identified topics are issues related to XML (12.69), the creation and customization of editors for EMF models (12.55), persistence and resources (12.49), and model annotation in the context of (the customization of) code generation (12.31). Not shown are topics on model repository issues involving, e.g., Connected Data Objects (CDO) (9.64), model transformations and model import and export (8.31).

### Topics in Xtext Forums (Table 3).

Although the Xtext category includes just one tool (Xtext), it boasts the second highest number of total posts (Figure 1).

Not surprisingly, most posts (24.0) deal with the creation of a Domain Specific Language (DSL) using Xtext or the implementation of an existing language in Xtext. Plugin issues are next (18.89), followed by problems with grammar

rules (13.47), more general, Eclipse-related installation and update issues (11.55), code generation issues (10.23), and issues arising from the customization or integration of an Xtext editor (9.85) or grammar export and import (9.32).

Topics found, but not shown in the table revolve around project build issues with Maven (8.94), the use and integration of Xbase, a language for Java expressions implemented in Xtext (8.81), and problems related to scoping and referencing in grammars (8.72). The first of these shows increasing trends.

Overall, our analysis suggests that Xtext users would benefit most from better support from Eclipse for plugin management and tool installation and updates. More Xtext-specific improvements should focus on improved support for grammars in general, and their definition, creation, use (e.g., via references), and migration (e.g., conversion of grammars in other formats to Xtext’s format and import/export) in particular.

### Topics in MTT Forums (Table 4).

The majority of posts on MTT forums relate to documentation or similar additional supporting sources of information that would help users to learn the tool and get started (44.0). Many users also take their questions about basic concepts and how to do something to the forums themselves (24.30). Posts related to bugs and errors also feature quite prominently (31.58), as do posts from users with issues when running simple transformation projects (15.34). Another topic involves EuGENia, a tool for the automatic generation of GMF models from an Ecore metamodel (9.89).

Not shown are topics related to the application transformation rules, especially for ATL (9.43), JET templates (8.50), the EVL constraint language (8.18).

Our analysis suggests that there is a need in the MTT community for more material that makes it easier for users to learn and start using a tool. A recent survey of model transformation tools [12] found not only a staggering number of tools, but also many which appear to have been discontinued or abandoned. Inadequate documentation can cause high barriers to entry which might prevent tool adoption.

### Topics in MDT Forums (Table 5).

In this category, the major topics of discussion involve bugs and errors, graphical user interface issues, and prob-

**Table 4: MTT Topics.**

Top LDA Words	Description	Share	Trend
<i>model code languag provid implement answer specif document read kind creat write inform featur tool</i>	Documents and information	44.70	
<i>problem error bug messag fix line report time chang code wrong found test attach version</i>	Bugs and Errors	31.58	
<i>model ecor metamodel emf gener instanc element xmi xml transform refer creat input load xsd</i>	Conceptual and ‘How to’ questions	24.30	
<i>eclips project plugin jar folder depend workspac build run import creat compil runtim path</i>	Plugin dependencies and configurations	16.69	
<i>oper element object method list collect call variabl queri sequenc iter paramet defin thi loop</i>	Model transformation language issues related to data structures such as collection, sequence, set	15.74	
<i>run transform model launch configur code creat gener execut project output select click eclips load</i>	how to create and run transformation project	15.34	
<i>gmf eugenia diagram editor gener label creat link annot node tool model target icon extend</i>	EuGENia issues, e.g., generating GMF models from Ecore models or transforming Ecore to GMF	9.89	

**Table 5: MDT Topics.**

Top LDA Words	Description	Share	Trend
<i>problem error bug fix version time chang line miss build wrong solv open messag expect</i>	Bugs and errors	22.46	
<i>diagram model creat editor uml select view element menu open add click explor packag show</i>	Graphical editor issues mostly related to Papyrus and UML2 (e.g., hyperlinks between two models)	19.8	
<i>uml model implement metamodel specif extend gener languag instanc provid custom understand read time tool</i>	Extending UML tools	18.47	
<i>uml ecor gener model code emf genmodel convert creat eclips meta-model project import map transform</i>	UML code generation and related transformations from Ecore to UML	14.96	
<i>papyru eclips model plugin version code project sysml camil custom extens provid palett instal mart</i>	Papyrus related issues, including installation and compatibility with Eclipse versions and source code	14.6	
<i>model project eclips uml tool emf code tutori creat develop vlad plugin inform revers start</i>	Requests for tutorials & documentation	10.7	
<i>eclips instal updat uml featur site download tool depend build sdk version requir emf plugin</i>	Update site issues and plugin dependency problems	10.32	
<i>profil stereotyp appli uml model defin creat static editor load problem applic packag definit save</i>	UML profile and stereotype issues	9.8	

lems with UML profiles and stereotypes. On the upside, the share of discussions on bugs and errors has recently begun to trend down suggesting possibly that improvements are beginning to take effect. Also, we notice that discussions involving Papyrus have been increasing sharply.

A discussion topic of joint interest to users of both MTT and MDT tools are issues with constraints (i.e., the expression of constraints in some language and the use of constraints for validation), reflecting the important role that constraints play in the definition of both models and model transformations. The three remaining topics not listed in Table 5 are the import/export of XML-based models with a share of 9.12, OCL constraints and OCLinCore related issues with share of 8.67, and loading and importing models in Ecore, MagicDraw, and XMI, with a share of 6.90.

### Topics in GFT Forums (Table 6).

In this category, discussions are primarily concerned with the quality of the documentation and tutorials (36.26), bugs (17.23), and basic activities such as editor creation (18.32) and the creation and modification of model elements (14.76). We note that GMF builds on EMF, which means users have to understand the relevant parts of EMF before they can start using GMF.

Additional topics found, but not listed in the table are

related to plugin dependencies, update and installation failures, the GMF Runtime, and customized shapes.

## 3.2 Research Question 2

*Which issues, problems, and challenges are “newbies” facing and how many of their questions end up being answered satisfactorily?*

### 3.2.1 Motivation

This part of our investigation is aimed at identifying so-called “barriers of entry”, i.e., issues that “scare off” interested people and prevent these potential users from turning into actual users. Once identified, these barriers could then be mitigated through, e.g., a new tool release or a simplified installation procedure, or more comprehensive or detailed supporting material. Moreover, these barriers might also help educators and instructors to better prepare students or employees for the use of modeling tools.

### 3.2.2 Results

Not surprisingly, there are similarities between the general topics discovered for RQ1 and the groups of newbie questions. This similarity lends some more credibility to the automatically discovered topics. In the following, the paragraph title is followed by the number of questions the

Table 6: GFT topcis.

Top LDA Words	Description	Share	Trend
<i>code gmf implement time start tutori provid gener gef kind edit cus- tom inform answer understand</i>	Documentation and tutorials	36.26	
<i>model element diagram creat editor domain ecor uml gmf problem refer instanc graphic metamodel shortcut</i>	Editor creation for Ecore and UML	18.32	
<i>problem error gmf diagram bug code eclips gener fix mindmap tutori messag editor run exampl</i>	Bugs in GMF and examples	17.23	
<i>node compart child parent children creat insid refer element map add top problem featur list</i>	Creating and editing model elements, e.g., node, compartment and their related issues	14.76	
<i>diagram element view model refresh chang updat creat semant notat problem edit edg node add</i>	Model view and presentation issues	13.36	
<i>model gmf editor emf gener xml ecor creat graphic format data dia- gram merk xsd save</i>	GMF model saving and loading from XML (model transformation)	10.92	
<i>method editpart call chang public overrid code event void protect se- lect thi notif view extend</i>	Graphical properties of models, e.g., size and color	10.77	

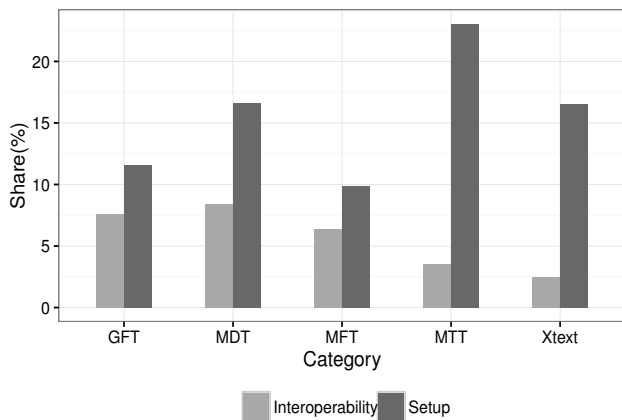


Figure 3: Groups of most common newbie questions.

results are based on and from which forums they came.

### Common Newbie Questions (2,339, all).

The most prominent groups of questions common to all tools are *setup issues* and *interoperability issues*.

1. *Setup issues*: This category contains questions pertaining to the download, installation, and basic use of a tool to, e.g., create, compile, build, or execute a project. The fact that novel users of all selected tools deal with setup issues is not surprising. However, the relative frequency of these questions differs to an unexpected extent (Figure 3): 23% of newbie questions on the MTT forums (Epsilon: 11.70%, ATL: 7.31%) are setup questions, compared with only 10% on the MFT forums.

2. *Interoperability issues*: This category contains questions about importing and exporting artifacts such as models or projects, and the conversion of artifacts produced by a different tool prior or during import. E.g., problems reported on the forums relate to the import of IBM RoseRT models to Papyrus, and the conversion of UML models to Ecore models. The relative significance of questions about interoperability among newbies suggests that many of them already have been using related tools and that they are looking for

ways to reuse existing artifacts or to use the tools jointly.

### MFT Newbie Questions (303, EMF & ECore).

1. *Code Generation (35%)*: Questions ask, e.g., how to initiate and automate the code generation, how to fix errors occurring during code generation, how to configure Genmodel parameters, how to keep the model and the generated code synchronized, and how to customize the generate code using annotations.

2. *Modeling (19%)*: Second most frequent were questions about EMF and how to use it to create models. Questions aim at general modeling and metamodeling concepts (e.g., containment and reference relationships, and derived attributes), the creation and design of models, and specific EMF concepts.

3. *EMF Persistence API (14%)*: These questions are concerned with the persistence API offered by EMF for saving (serializing) and loading (de-serializing) models. Problematic concepts mentioned include Resource, ResourceSet, ResourceFactory, and URIConverter.

4. *Use Generated Code for the Model (12%)*: Next are questions about the use of code generated from the model. Posts ask about, e.g., the use of factories for creating models, and how to program modifications to existing models.

5. *Editor (11%)*: Most questions in this group pertain to EMF.Edit, a framework for the creation of editors for EMF-based models.

### Xtext Newbie Questions (362, Xtext).

1. *Grammars (47%)*: Almost half of all newbie questions deal with grammars. Closer inspection reveals that Xtext's use of cross referencing is causing the most trouble. In Xtext, cross references can appear in the grammar to, e.g., allow for non-containment references. Just like in traditional languages, a linker would then resolve these cross references, possibly with the help of the scoping API. Other challenges include the handling of datatypes, quotation, comments, and hidden rules in Xtext grammars, as well as grammar import.

2. *Editor (17%)*: These questions target the customization of the generated editor and its integration with other applications, as well as hovering and error checking.

3. *Use of Generated Code (16%)*: Questions are about the use of the generated code for validation and code generation (from the language defined with Xtext) tasks. Questions



ask, e.g., how to add an import list to generated code, and how to map an Xtext model to Java code.

4. *General Questions (9%)*: Questions deal with the general capabilities and features Xtext such as supported migration methods from existing languages to Xtext, ways for designing a language from scratch, code generation, and integrating Xtext with existing compilers.

5. *Scoping (8%)*: Xtext’s Scoping API allows the definition of how non-containment references are resolved. Understanding scoping appears to be challenging for new users.

### *MDT Newbie Questions (296, Papyrus).*

Papyrus is a highly customizable MDE environment supporting a range of languages and standards such as UML, SysML, OCL, FUML, and ALF (MARTE, EAST-ADL, RobotML, and UML-RT are in incubation), the definition of Domain Specific Languages, and model-based activities such as simulation, testing, and safety analysis. Many parts of Papyrus can be customized including the UML profile, the model explorer, diagram notation and style, views and menus.

1. *User Interface (UI) (48%)*: The majority of questions pertain to UI difficulties including setting or viewing properties, drawing diagrams, connecting nodes, hiding/showing elements in diagrams, configuring viewing points, and defining and applying style sheets.

2. *Profile and Stereotype (20%)*: This group contains questions about defining and applying the UML profile. Questions ask about, e.g., how to define flowcharts using profiles, and how to switch between local profiles contained in the workspace and profiles registered in the runtime.

3. *Code Generation and Validation (11%)*: Questions focus on existing code generation features, adding a new code generator, and adding constraints using OCL.

4. *Model Simulation (4%)*: A few questions relate to model simulation using the Moka plugin or other tools.

### *MTT Newbie Questions (205, ATL&Epsilon).*

1. *Coding Help (58%)*: A large number of questions request code snippets or guidance on how to fix specific coding issues. We remark that while most other tools are based on Java, most MTT tools offer one or even several specific languages to specify transformations [12]. For instance, ATL defines its own language which is based on OCL; Epsilon is a family of a total of eight languages including the Epsilon Transformation Language (ETL) for model-to-model transformation, the Epsilon Generation Language (EGL) for code generation, the Epsilon Validation Language (EVL) for expressing constraints for model validation. Consequently, users need to learn the languages associated with the tools before using them.

2. *General Questions (33%)*: General questions about model transformation concepts and how to approach specific transformation tasks account for a third of questions. Questions ask about, e.g., how to transform XMI to Ecore, and rule scheduling for ETL.

### *GFT Newbie Questions (183, GMF&Graphiti).*

1. *Editor Customization and Graphical Display (55%)*: More than half of the questions are about configuring or customizing an editor, or displaying model elements graphically. More specifically, questions focus on how to add elements to the model programmatically, how to customize the

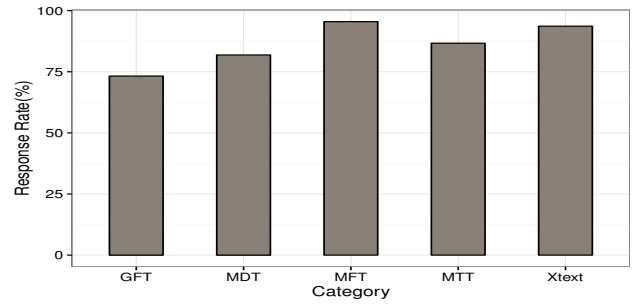


Figure 4: Response rate of newbie questions.

palette, how to drag and drop in the editor, how to layout the diagrams, how to update and synchronize diagrams, how to draw customized shapes such as polygons, how to change the appearance of nodes and shapes, and how to add text to connections and nodes.

2. *General Questions (26%)*: These questions ask about which features are supported by the tool, whether the tools are suitable for the user’s needs, how the tool can be integrated with other modeling tools, and how to work with non-EMF models. Moreover, requests examples and tutorials are made.

3. *Code generation and Model Validation (6%)*: Questions are concerned with customizing code generation, solving specific code generation problems, and validating the models the user creates.

### 3.2.3 Results of Response Rate Analysis (2,339, all)

We define the response rate of a forum as the percentage of the newbie questions that are eventually answered satisfactorily. We consider a response to have answered a question satisfactorily if it provides an answer known to be correct, or if it is followed by some kind of positive acknowledgment from the person who had asked the question.

Figure 4 shows the response rate of each category. The results show that the overall, average response rate is quite high (86%). The response rates of MFT, Xtext, and MTT sit above this average with 95.5%, 93.64%, and 86.68%, respectively. Within MTT, the rate for Epsilon is an astonishing 100%, while for ATL it is 69%. On the low end, MDT achieves 81.87% and GFT has the lowest response rate with 73.23%. Within GFT, less than 40% of newbie questions about GMF ended up being answered.

## 4. THREATS TO VALIDITY

In this section, we discuss limitations that may threaten the validity of our results.

### *Data Extraction, Cleansing and Automatic Topic Extraction (RQ1).*

To minimize the risk of coding errors we used the existing libraries [19, 1] for data extracting and cleaning. Also, the way we prepared the data for LDA follows standard practice and there is some evidence that indicates the usefulness of these steps [4]. Similarly, we followed suggestions in the literature with respect to choice of the LDA parameters combined with manual experimentation, but it is possible that a different choice of parameters, most importantly number of topics  $K$  and the threshold  $\delta$ , would have produced dif-

ferent results. Overall, however, we believe this difference to be very slight. For instance, the choice of  $\delta$  can impact the share of a topic; however, since  $\delta$  is applied to all topics, the utility of the share as an indication of the relative frequency of a topic remains intact.

### *Manual Topic Labeling (RQ1).*

Since this step requires human judgment and, on occasion, deep knowledge of a tool, inaccuracies are possible. For instance, it is possible that the description we gave an extracted topic is not as accurate as it could be and does not really capture the “essence”, if any, of the topic. To mitigate this problem, we always validated the description of a topic on as large a sample of the discussions with a high share in that topic as possible.

### *Manual Classification (RQ2).*

Our definition of a newbie question is not perfect: E.g., a long-time user may appear to us as a newbie (due to, e.g., having several accounts or never having joined a forum). In these cases, the purpose of a question from such a user would not be to help her overcome a “barrier to entry” and its content would not be representative of problems new users have. To mitigate this problem, we also check the content of questions and filter out clear, non-newbie questions. While a risk of missing or misclassified question remains, we believe it is low enough for our results to still provide an indication of central newbie issues.

## 5. RELATED WORK

The study of Question & Answer (Q&A) websites such as Stack Overflow and Yahoo has been an active topic of research in recent years. We will first present mining work with no particular connection to modeling, followed by work focusing on modeling in some form.

Our work is inspired by the growing body of research on mining software repositories [10]. This research has led to some relatively mature techniques and tools together with a good understanding of their strengths and weaknesses [15]. These techniques have already been used to mine discussion forums hosted by, e.g., Stack Overflow and Yahoo to identify developer concerns [25, 27, 9]. However, to the best of our knowledge, they have not yet been applied to Eclipse modeling forums.

Barua *et al.* [4] proposed a semi-automatic approach to analyze the textual content of developer discussions. The authors use LDA to discover the main topics present in Stack Overflow. Their findings indicate several relationships between topics and trends over time. Similarly, Treude *et al.* [25] also study data from Stack Overflow discussions. They analyze 385 questions and assign them into 10 categories. The authors observe that Q&A websites are particularly useful for code reviews and conceptual questions.

Topic modeling techniques are also used in [16] to discover important topics in mobile-development related questions on Stack Overflow.

Gyöngyi *et al.* [9] analyze 10 months of data from Yahoo Answers. The authors use the data to make several observations about the interests of users and the impact of responses. Adamic *et al.* [2] also analyze Yahoo Answers to cluster the top-level categories according to content characteristics and patterns of interaction among the users.

There is not much work on mining repositories relevant to the modeling community. Williams *et al.* [29] analyze the activity of the newsgroups of 10 Eclipse modeling projects. For each project, their findings show how different metrics, such as the number of new users and active users, the number of new threads, and the number of messages classified as requests and replies, have evolved over time.

Kolovos *et al.* [13] study GitHub to analyze the use of Eclipse-based MDE technologies in open-source software development projects. They observed a large number of GitHub repositories (1928) using these technologies, and a substantial community of MDE developers (2195).

We conclude that, to the best of our knowledge, no attempt has been made to use Eclipse forums to understand the problems of users of Eclipse modeling tools.

## 6. CONCLUSION AND FUTURE WORK

We have presented the results of our semi-automatic and manual analyses of the posts on the forums of Eclipse modeling tools. The analyses aim at determining which kinds of issues and problems general and novice users bring to these forums and, for novice users, how often satisfactory responses are provided. Our results provide supporting evidence for some already commonly held, but unproven, beliefs, as well as some interesting new insights. For example, complaints about Eclipse’s plugin architecture, the quality of documentation, and the support for initial tool setup and interoperability may not come as a surprise, but our work provides some concrete evidence, possibly useful for convincing decision makers in granting agencies and businesses. However, our study also suggests that research into, e.g., the classification of Eclipse plugin issues; the conversion, migration and manipulation of grammars (for Xtext); the role of constraints in modeling; and the integration and customization of generated artifacts such as models, code and editors (MFT, Xtext, and GFT) might advance the state-of-the-art in Eclipse-based modeling. Moreover, some concrete, specific suggestions about how a tool might be improved are given in Section 3.

Apart from the observations and results the study provides, the contributions of our work also include the data that our analyses are based on: all forum data mentioned in the paper is publically available [11].

We have been careful not to “read too much” into the results of our study. E.g., as discussed at the beginning of Section 3, it is tempting to interpret declining numbers of discussions in a forum as declining interest in the tool, or to link frequent requests for better documentation to a perceived lack of adoption of a tool. More work is necessary before conclusions about what impacts the adoption of tools can be made. We welcome and encourage the use of our data set for similar studies. Much more work is needed to better understand how modeling tools and their supporting ecosystems can be improved and which role the quality of tools plays in the adoption and transfer of technology in general and MDE in particular.

## 7. ACKNOWLEDGMENTS

This work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and by the Ontario Ministry of Research and Innovation (MRI).

## 8. REFERENCES

- [1] Stop words. <https://pypi.python.org/pypi/stop-words>, 2015.
- [2] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman. Knowledge sharing and Yahoo answers: everyone knows something. In *17th International Conference on World Wide Web (WWW'13)*, pages 665–674, April 2013.
- [3] K. Barnard, P. Duygulu, D. Forsyth, N. D. Freitas, D. M. Blei, and M. I. Jordan. Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135, November 2003.
- [4] A. Barua, S. W. Thomas, and A. E. Hassan. What are developers talking about? An analysis of topics and trends in Stack Overflow. *Empirical Software Engineering*, 19(3):619–654, June 2014.
- [5] T.-H. Chen, S. W. Thomas, and A. E. Hassan. A survey on the use of topic models when mining software repositories. *Empirical Software Engineering*, 2016. 77 pages. In print.
- [6] J. Den Haan. 8 reasons why model-driven approaches (will) fail. <http://www.infoq.com/articles/8-reasons-why-MDE-fails>, 2008.
- [7] Eclipse. Eclipse modeling community. <https://www.eclipse.org/forums/index.php/i/5/>, 2016.
- [8] T. L. Griffiths and M. Steyvers. Finding scientific topics. In *National Academy of Sciences*, pages 5228–5235, April 2004.
- [9] Z. Gyongyi, G. Koutrika, J. Pedersen, and H. Garcia-Molina. Questioning Yahoo! answers. In *1st Workshop on Question Answering on the Web*, 2007.
- [10] A. E. Hassan. The road ahead for mining software repositories. In *Frontiers of Software Maintenance*, pages 48–57. IEEE, October 2008.
- [11] N. Kahani, M. Bagherzadeh, J. Dingel, and J. R. Cordy. Dataset. <https://bitbucket.org/mojil/eclipsemodelingdata/downloads>, 2016.
- [12] N. Kahani and J. R. Cordy. Comparison and evaluation of model transformation tools. Technical Report 2015-627, December 2015.
- [13] D. S. Kolovos, N. Matragkas, L. Korkontzelos, S. Ananiadou, and R. Paige. Assessing the use of eclipse mde technologies in open-source software projects. In *Workshop on Open Source Software for Model Driven Engineering (OSS4MDE'15)*, 2015.
- [14] A. Kuhn, G. Murphy, and C. Thompson. An exploratory study of forces and frictions affecting large-scale model-driven development. In *15th International Conference on Model-Driven Engineering Languages and Systems (MODELS'12)*, number 7590 in LNCS, pages 352–367, 2012.
- [15] I. Kwan and D. Damian. A survey of techniques in software repository mining. Technical Report DCS-340-IR, University of Victoria, 2011.
- [16] M. Linares-Vásquez, B. Dit, and D. Poshyvanyk. An exploratory analysis of mobile development issues using Stack Overflow. In *10th Working Conference on Mining Software Repositories (MSR'13)*, pages 93–96, May 2013.
- [17] C. D. Manning, P. Raghavan, and H. Schütze. *An algorithm for suffix stripping*. Cambridge University Press, New York, 2008.
- [18] A. McCallum. Mallet: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2016.
- [19] M. Porter. The porter stemming algorithm. <http://tartarus.org/martin/PorterStemmer>, 2016.
- [20] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, March 1980.
- [21] M. Staron. Adopting model driven software development in industry – a case study at two companies. In *9th International Conference on Model-Driven Engineering Languages and Systems (MODELS'06)*, volume 4199 of LNCS, pages 57–72, 2006.
- [22] S. W. Thomas. Mining software repositories using topic models. In *33rd International Conference on Software Engineering (ICSE'11)*, pages 1138–1139. ACM, May 2011.
- [23] S. W. Thomas, B. Adams, A. E. Hassan, and D. Blostein. Studying software evolution using topic models. *Elsevier*, 80(B):457–479, February 2014.
- [24] F. Tomassetti, M. Torchiano, A. Tiso, F. Ricca, and G. Reggio. Maturity of software modelling and model driven engineering: A survey in the italian industry. In *16th International Conference on Evaluation and Assessment in Software Engineering (EASE'12)*, volume 4199 of LNCS, pages 91–100, 2012.
- [25] C. Treude, O. Barzilay, and M.-A. Storey. How do programmers ask and answer questions on the web? In *33rd International Conference on Software Engineering (ICSE'11), NIER track*, pages 804–807. IEEE, May 2011.
- [26] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimnok. Evaluation methods for topic models. In *26th Annual International Conference on Machine Learning (ICML'09)*, pages 1105–1112. ACM, June 2009.
- [27] S. Wang, D. Lo, and L. Jiang. An empirical study on developer interactions in StackOverflow. In *28th Annual ACM Symposium on Applied Computing (SAC'13)*, pages 1019–1024. ACM, March 2013.
- [28] J. Whittle, J. Hutchinson, M. Rouncefield, H. Burden, and R. Heldal. Industrial adoption of model-driven engineering: Are the tools really the problem? In *16th International Conference on Model-Driven Engineering Languages and Systems (MODELS'13)*, pages 1–17. Springer, 2013.
- [29] J. Williams, N. Matragkas, D. Kolovos, S. Ananiadou, and R. Paige. Software analytics for mde communities. In *Workshop on Open Source Software for MDE (OSS4MDE'14)*, 2014.