Collaborative Model-Driven Software Engineering: A Systematic Update

Istvan David*, Kousar Aslam†, Sogol Faridmoayer*, Ivano Malavolta†, Eugene Syriani*, Patricia Lago†‡

*DIRO, Université de Montréal, Canada – {istvan.david, sogol.faridmoayer}@umontreal.ca, syriani@iro.umontreal.ca

†Vrije Universiteit Amsterdam, The Netherlands – {k.aslam, i.malavolta}@vu.nl

‡Chalmers University of Technology, Sweden – p.lago@vu.nl

***Abstract*—Current software engineering practices rely on highly heterogeneous and distributed teams working in a collaborative setting. Between 2013–2020, the publication output in the field of collaborative Model-Driven Software Engineering (MDSE) has significantly increased. However, the only systematic mapping study available is limited to studies published until 2015. In this paper, we provide an update on that study for the complementing 2016–2020 period, and report the latest results, challenges, and trends. Our analysis led to selecting 29 clusters of 54 new peer-reviewed publications on collaborative MDSE. Based on the novel developments in the field, we have extended and improved the original classification framework, making it applicable to recent and future research contributions on collaborative MDSE. The insights in this paper relate to the changing trends in the field and present new relevant information.**

***Index Terms*—Model-driven engineering, collaborative modeling, systematic mapping study, systematic update**

I. Introduction

Current software engineering practices rely on highly heterogeneous and distributed teams, required to work together to deliver the software system correctly and efficiently. Thus, collaboration across team members, and often across multiple teams, is needed [1]. The combination of computer-aided collaboration, and model-driven software engineering (MDSE) presents its own benefits and challenges [2]–[4]. In the past decade, collaborative MDSE has become a prominent feature of today's software engineering practice, e.g., in agile methodologies and low-code platforms [5]–[7].

The field of collaborative MDSE is rapidly expanding and maturing. The only comprehensive study in the field has been provided by Franzago et al. [8][1], encompassing the 20-year period of 1996–2015. Considering the recent improvements in collaborative MDSE, however, we found that a systematic update of the study on the 5-year period of 2016–2020 will provide valuable insights. Apart from the intensive academic research, this period witnessed various editions of workshops on collaborative modeling [9] [10], a thematic special issue on the topic [4], and numerous international research and development projects, most notably [11]–[13].

In this paper, we report the results of our systematic update. By following the guidelines of Mendes et al. [14], we found 54 relevant new primary studies. On average, this accounts for 10 studies per year, twice as much as the 5 studies per year measured by the original study. Although these numbers might already hint to a need for an update, we further motivate our work by a systematic qualitative assessment. Additionally, we have surveyed the field for secondary studies to enhance the classification framework of the original study. Eventually, three secondary studies were considered, out of which we retained [15], as it complemented the original classification with relevant information.

This update is strongly coupled with the original study. It takes assumptions from it, and follows its slightly adapted methodology [16]. Accordingly, the goals of this paper are aligned with those of the original study: to identify, classify, and understand collaborative MDSE approaches that have emerged since the original study, i.e., between 2016–2020. An additional goal of this update is to enhance the classification framework of the original study, in response to the developments in the field. Our research is not limited to tools or technological stakeholders. Instead, we approach the research questions from a holistic standpoint. However, we restrict our investigation to approaches which are (i) model-driven (or at least model-based); (ii) consider a meaningful set of collaborating stakeholders; and (iii) provide appropriate means of communication.

There are two kinds of information reported in this paper. First, we reflect on the takeaways and trends of the original study, and report any changes from the 2016–2020 period. Second, we identify new trends based on the newly sampled papers, and the newly introduced classification categories.

[1]Referred as the *original study* in the remainder of the paper.

II. Related Work

The only systematic review on collaborative MDSE is the original study of this update by Franzago et al. [8]. They rigorously reviewed and analyzed 48 primary studies collected until mid-2015. In this systematic mapping study, they elaborated a classification framework along three main dimensions. *Model management* is the dimension to manage the life cycle of models in a collaborative setting through, for example, repositories to persist models, modeling tools to manipulate models, and interchange formats to share models across stakeholders. *Collaboration* is the means for stakeholders to collectively work on models and coordinate themselves with, for example, versioning systems, conflict management systems, model comparison engines, and development and managerial processes. *Communication* ensures that stakeholders of each other's work can exchange and interact with each other (e.g., by sharing design decisions, tracking discussions, and notifying changes). The mapping study details a taxonomy for each dimension from which they draw various conclusions on the topic of collaborative MDSE. In this paper, we update the original study with numerous papers that have appeared since then. Methodologically, we meticulously follow their protocol, but also update the classification framework with new characteristics that have appeared since then.

While retrieving publications for this update, we identified three secondary studies related to collaborative MDSE. Masson et al. [15] performed a feature analysis of over 10 collaborative modeling tools. They discuss various features already supported by existing tools and identify unavailable, yet desirable, collaborative features. Since their study is well-aligned with our scope, we have included the relevant charactersitics in our classification framework. Stephan [17] reports temporal trends of keywords in 103 collaborative MDSE papers between 2012–2017. Although this information is not sufficient to incorporate in our study, we confirm that all the top keywords found appear in our data extraction form. Ertugrul and Demirors [18] analyzed three role-based collaborative business process modeling approaches. Their work is specific to process modeling which is a narrower scope than the one we are concerned with in this paper.

Many modeling tools have recently shifted to support different forms of collaboration. Just to name a few, AToMPM [19] provides a purely in-browser interface for multi-paradigm modeling activities. WebGME [20] also supports web-based collaboration where users can work on the same model thanks to a branching scheme similar to Git. MetaEdit+ [P13] has also released a collaborative environment that incorporates Git to provide offline collaboration. GenMyModel [21] provides an in-browser client to model collaboratively with various model management utilities. OBEO Designer [22] also enables collaborative modeling within Eclipse with a locking mechanism.

III. Assessing the Need for an Update

Mendes et al. [14] report that many systematic studies are potentially outdated; thus affecting the aggregated understanding of the state-of-the-art through those systematic studies. In order to facilitate a factual decision whether an update of a study is required, they provide a systematic assessment framework. In this section, we apply the framework to our case, and report the results of the assessment.

The framework defines a three-step decision process. In the first step, we must assess if the original study is still of current interest by answering three specific questions. If we can answer positively to all three questions, then we can proceed to the second step. Here, we identify whether new relevant methods, studies or information are available with respect to the topic of the original study by answering two specific questions. If at least one is answered positively, we can proceed to the third step. There, we assess if updating the review has an effect on the findings, conclusion or credibility of the original study by answering two specific questions. If at least one is answered partially positively, the original study is deemed as a good candidate for an update. The remainder of this section demonstrates with quantitative evidence that there is a clear need to update the original study [8].

A. Assessing the currency of the original study

1) Does the original study still address the current question?: Since the publication of the original study, three editions of a workshop on collaborative modeling [9] have been organized between 2016–2018. A special issue on collaborative MDSE has been published in the IEEE Software journal in 2018 [4]. A recent article [23] firmly places collaborative modeling as one of the future grand challenges in MDSE. Another recent article [1] suggest that collaborative features will retain their prominent role in modeling environments in the future. Therefore, we confirm that collaborative MDSE is still a current mainstream topic in the community.

2) Has the original study had good access or use?: Mendes et al. suggest that a secondary study should be cited at least 6 times per year to be considred for an update. This is based on [24], where the authors report that software engineering papers are cited on average of 6.82 times per year ($n = 71\ 668$). At the time of writing, the original study has accumulated 47 citations over the 2-year period since its publication, resulting in 23.5 citations per year. This ranks the original study in the top third of the list of studies investigated by Mendes et al. Furthermore, the original study is currently the highest-cited publication by the search term *"collaborative model-driven"* in all major scientific databases. Therefore, we confirm that the original study has had a good access and use.

3) Has the original study used valid methods and was well conducted?: The original study rigorously followed the guidelines to conduct systematic mapping studies [25], [26], and scored above average, at 61.5%, in the quality checklist of Petersen et al. [27]. Therefore, we confirm that the methods of the original study are well-constructed and executed.

B. Relevant new methods, studies and other information

1) Are there any new relevant methods?: As outlined in Section II, new studies have been published since the publication of the original study. In particular, the work of Masson et al. [15] has a relevant classification method complementary to the original one. The authors provide a feature model for collaborative modeling environments, focusing on the implementation details of such systems. The feature model could serve as an aid for extending and updating the classification framework of the original study.

2) Are there any new studies, or new information?: We have evaluated the search string of the original study[2] over the time range the study has focused on (1996–2015), and over the time range since then (2016–2020). The search string

[2] `(collaborat* OR coordinat* OR cooperat* OR concur* OR global) AND (MDE OR MDD OR MDA OR MDS* OR EMF OR DSL OR DSML OR "model driven" OR "eclipse modeling framework" OR "domain specific language" OR "domain specific modeling language")`

yields 17 300 hits for the original 20-year range and nearly as much, 16 800 hits, for the 5-year range of 2016–2020. This is a substantial increment in annual publications by a factor of 3.88. As this study update shows, a significant amount of work has been dedicated to collaborative modeling following the publication date of the original study.

C. Assessing the effect of the update

1) Will the adoption of new methods change the findings/conclusions or credibility?: The adoption of the feature model of [15] alone could improve at least the credibility of the findings, as it provides a complementary technical set of information to the original study. Other studies might be encountered during the search phase that could further influence the findings and conclusions.

2) Will the inclusion of new studies/information/data change the findings/conclusions or credibility?: Multiple papers on advanced collaborative techniques have been published since the publication of the original study. Typical topics include property-based locking [28], semantic inconsistency management [29], and other advanced techniques [P01]. Therefore, we are confident that the inclusion of new studies will most probably update and extend the classification framework drawn in the original study.

Given the positive answers above, we deem it justified and required to update the original study. As a rule of thumb, Mendes et al. [14] suggest that the maturation time between the publication of the original study and its update should be longer than two years. In our case, this time is 5 years.

IV. Study Design

Our study was carried out in accordance with well-established guidelines in the realm of empirical software engineering [26], systematic mapping studies [27], and updates of systematic literature studies [30], [31]. For the sake of consistency, we keep the design of this study aligned with the one of the original study as much as possible (For example, by using the same research questions, same selection criteria, etc.). In the following, we give an overview of the design of this study, emphasizing its variation points with respect to the original one. Further details about the common parts can be found in the original study. A complete *replication package* is publicly available [32] for independent replication and verification of our study. The replication package includes the raw data of our search and selection phase, the list of selected primary studies, the raw data extracted from each primary study, the R scripts for data exploration and analysis, and the list of changes in the classification framework.

A. Goal and Research questions

We formulate the goal of this study using the Goal-Question-Metric perspectives [33]. Accordingly, the goal of this study is to: identify, classify, and understand issues related to the characteristics, challenges, and publication trends of collaborative MDSE approaches, in the period between 01.01.2016–31.12.2020, from a researcher's point of view.

We use the same research questions as the original study.

RQ1: *What are the characteristics of collaborative MDSE approaches?*

RQ2: *What are the challenges and shortcomings of existing collaborative MDSE approaches?*

RQ3: *What are the publication trends that can be deduced from the scientific publications about collaborative MDSE approaches over time?*

By answering these research questions we provide an *up-to-date map* that classifies recent approaches for collaborative MDSE with respect to: (i) their model management, collaboration, and communication characteristics; (ii) their limitations, faced challenges, and future work; and (iii) publication trends such as publication year, and targeted scientific venues. Our map allows current and prospective researchers to understand the evolution of this research field in the past years, thus allowing the scientific community to better reason about research interests, and unexplored research directions. Figure 1 outlines the process and main phases of this study.

B. Search and selection

The success of a systematic study strongly depends on the retrieval of the relevant primary studies that are representative enough of the topic being considered [25]. As recommended in the guidelines for updating a systematic literature study by Wohlin et al. [30], we have based our search strategy entirely on forward snowballing, starting from (i) the set of primary studies of the original study, and (ii) the original study itself. Although the authors suggest one iteration is sufficient for mapping the increment of the state of the art, we have opted for a fully recursive forward snowballing to improve the quality of the search. In accordance with [30], we relied on Google Scholar as the source of retrieval. Google Scholar is considered a comprehensive academic search engine [34], and adequate for updates of systematic literature studies [30].

As shown in Table I, in the first iteration of snowballing, we obtained a total of 631 potentially relevant studies. For each study, we have applied a set of selection criteria. If a paper was included, snowballing was applied iteratively on it. The procedure eventually concluded after four iterations. By this phase, we analyzed a total of 886 potentially relevant studies. We also found three papers from 2015 that were missing from the original study, due to their late publication date. These papers are considered in the analysis of long term publication trends (Section VII).

TABLE I: Statistics of the snowballing rounds

Snowballing	All	Excluded	Included
1st iteration	631	592	39 (6.18%)
2nd iteration	228	214	14 (6.14%)
3rd iteration	26	25	1 (3.85%)
4th iteration	1	1	0
Total	**886**	**832**	**54 (6.09%)**

The selection criteria are identical to the ones of the original study. We included studies that propose an MDSE approach

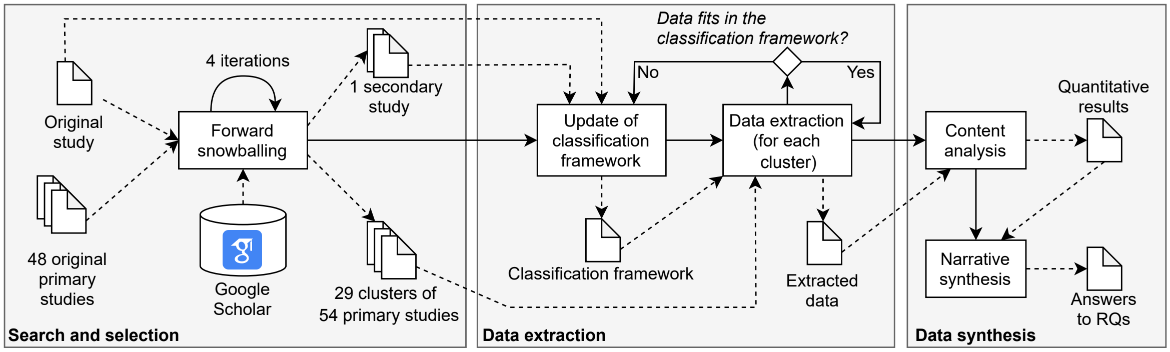

Fig. 1: The process of updating the systematic mapping study.

for supporting the collaborative work of multiple stakeholders. The study has to cover the three complementary dimensions of model management, collaboration, and communication. The primary artifacts within the collaboration process must be models. The studies should provide either validation or evaluation. We only considered studies written in English, available in full-text and subject to peer-review. We excluded studies that discuss only business processes and collaboration practices, without proposing a specific method or technique. Secondary studies and other forms (e.g., tutorials, extended abstracts, posters, editorials) were also excluded.

We used ReLiS [35] to perform the iterative screening of the studies among three researchers. Each of the 886 studies were screened by two researchers independently. For each potentially relevant study, we applied an *adaptive reading depth* [27], [36]. The resulting 0.86 Cohen's kappa indicates a nearly perfect agreement among the researchers. This makes us reasonably confident about the objectivity of our selection phase. Among the 886 studies, only 69 conflicts (7.8%) have emerged. We resolved each conflict by including the third researcher, and engaging in a hands-on discussion and collective decision making exercise. We have anticipated the first iteration having the most impact on the eventual set of included papers, and therefore, at the end of the first iteration, a fourth researcher validated all included studies, and 20% of the excluded ones. Eventually, we included 54 primary studies.

Consistent with the original study, we clustered the included studies in order to group the papers pertaining to the same approach. (For example, a conference paper extended into a journal version.) In accordance with [26], usually it was the most recent journal version of each cluster that was selected to represent the cluster during the investigation of RQ1 and RQ2. In the analysis of RQ3 (publication trends), however, we used each study separately. In the end, we grouped the 54 primary studies into a total of 29 clusters. For independent verification, the raw data related to each phase of our search and selection process is available in the replication package.

C. Data extraction

The goal of this phase is to extract data from each primary study to properly answer our research questions. Following the guidelines [14], we built on the data extraction procedure from the protocol of the original study. Specifically, we started from the same classification framework of the original study. Then, in order to accommodate the recent developments in collaborative MDSE, we updated the classification framework based on the secondary studies we have identified during snowballing. The only relevant secondary study was the feature model of collaborative modeling tools by Masson et al. [15]. Eventually, we included 11 new categories (one in the model management dimension, 9 in the collaboration dimension, and one in the communication dimension), and updated 7 categories of the original study by extending and aligning their parameters with the ones defined in [15]. Three researchers extracted the data from primary studies according to the updated classification framework; and the data from primary studies of the original study for newly defined categories. A fourth researcher validated the extracted data and resolved any conflicts. For independent verification, the complete classification framework and the extracted data are available in the replication package of this study. The resulting classification framework is presented in Sections V, VI, and VII.

D. Data synthesis

In this phase, we extracted and reported the main findings emerging from the classification framework. We performed a combination of *content analysis* and *narrative synthesis* [25]. Content analysis relies on the quantitative assessment of the extracted data (e.g., the frequency of web-based vs. desktop-based modeling editors), while narrative synthesis refers to the systematic method where a textual narrative summary is adopted to explain the quantitative information emerging from the content analysis and identifying emerging patterns and trends [25], [37], [38]. All authors collectively carried out this phase via a series of dedicated group sessions. We grouped the results of our data synthesis according to the research questions. These are reported in the following section. Furthermore, similarly to the original study, we used contingency tables to find new orthogonal insights which span multiple categories. The orthogonal findings are reported in Section VIII.

V. CHARACTERISTICS (RQ1)

In this section, we report the results of our study, with respect to the three dimensions of collaborative MDSE: model management, communication, and collaboration.

TABLE II: Support for Model Management

Value	Original (2003–2015)	Update (2016–2020)	Δ
Supported modeling artifacts			
Model	48 (100%)	29 (100%)	→
Metamodel	4 (8%)	9 (31%)	↑
Meta-metamodel	-	1 (3%)	↗
Language independence			
Independent	21 (44%)	24 (83%)	↑
Specific	27 (56%)	5 (17%)	↓
Validation			
Supported	15 (31%)	15 (52%)	↑
Editor type			
Graphical	38 (79%)	18 (62%)	↓
Textual	5 (10%)	6 (21%)	↑
Tabular	2 (4%)	5 (17%)	↑
Tree-based	14 (29%)	4 (14%)	↓
Sketch-based	7 (15%)	1 (3%)	↓
External	6 (13%)	1 (3%)	↓
Modeling framework – New category.			
Custom	40 (83%)	21 (72%)	↓
EMF	8 (17%)	8 (28%)	↑
Client type			
Desktop	31 (65%)	15 (52%)	↓
Web	18 (38%)	14 (48%)	↑
Mobile	-	4 (14%)	↑
Multi-view scenarios – New category.			
Multi-user single-view	38 (79%)	19 (66%)	↓
Multi-view single-model	7 (15%)	5 (17%)	↗
Multi-view multi-model	3 (6%)	5 (17%)	↑
Single-view multi-model	-	1 (3%)	↗
Multi-View support			
Synthetic	6 (13%)	5 (17%)	↗
Projective	8 (17%)	5 (17%)	→

↑ Increase ↗ Slight increase → No change ↘ Slight decrease ↓ Decrease

Data from original study. Data from update.

A. Model management

The model management dimension is responsible for managing the lifecycle of models, including their creation, manipulation, and storage. Table II lists the categories of the extended classification framework, the frequency of their values in the original study, the frequency of their values in this update, and an indication of the difference between the two studies. For the new categories, we manually extracted the corresponding data from the corpus of the original studies; for the previously existing categories, we use the data from the original study.

The focus of collaboration in terms of the *supported modeling artifacts* has shifted in 2016–2020 towards the: metamodel (9 occurrences) and meta-metamodel (1) level. With this shift, collaborative techniques have become more *language independent* as well. The ratio of language-independent techniques has nearly doubled, from 44% to 83%. *Validation* has become a more integral part of collaborative frameworks. More than half of the sampled approaches support the user with some form of validation, showing an increase from 31% to 52% since the original study. The lack of validation can significantly hinder collaboration, especially at higher levels of abstraction [39]. The increased support for validation, therefore, could be a result of the increased number of tools supporting collaborations at higher meta-levels.

The most notable changes in *editor types* are the decrease in use of graphical editors (from 79% to 62%), and the increased number of textual (from 10% to 21%) and tabular editors (from 4% to 17%). The vast majority of approaches, more than 70%, use their own custom *modeling frameworks*. For example, authors used a custom framework in [P13] to combine multi-user modeling with any external version control system without performing any merging or locking. The most widely used generic modeling framework, EMF [40] has improved its support from 17% to 28%.

We observed relevant changes in the *type of client software* for collaborative modeling. The ratio of desktop clients has decreased (from 65% to 52%) in favor of frameworks that provide either web (48%) or mobile client interfaces (14%). This shift highlights how users are becoming more mobile, and modeling is becoming less resource-intensive. For instance, [P26] presents a tool that facilitates the creation and use of graphical DSLs on mobile devices; the mobile-based editor was applied in a case study on wind turbines control applications development where the modeling activities were performed on-site in a wind farm.

Regarding the *multi-view scenarios* in collaborative approaches, the majority, nearly two third of the approaches operate in a multi-user single-view fashion [41], while true multi-view scenarios are supported only by a third of the frameworks. Nevertheless, multi-view approaches still have become more widespread overall. While only 21% of the approaches in the original study support multiple views, this has increased to 34% between 2016–2020, especially due to the strong increase in the ratio of multi-view multi-model techniques [42]. There is no real change in the trend of the ratio of synthetic (views built with different concrete syntaxes) and projective (views built with same concrete syntaxes) approaches [42] for multi-view support.

B. Collaboration

Collaboration features are responsible for enabling an effective and efficient groupwork across the involved stakeholders. Typical means of collaboration in MDSE include versioning systems with merging and branching support, consistency management mechanisms and conflict resolution algorithms.

As shown in Table III, real-time collaboration ("synchronous", in the original study) is becoming increasingly popular as compared to off-line ("asynchronous", in the original study) *types of collaboration*.[3] This could be the result of the increased use of web- and mobile clients, which align well with real-time principles, such as in the case of [P05], [P18] and [P22]. There is noticeable decrease in the ratio of approaches that operate over strictly *prescribed workflows* (from 90% to 52%). This means collaboration becoming less sequential, and featuring less distinctively defined roles. The three typical *consistency models* are still the strong, the eventual, and strong eventual models. Strong consistency ensures the identical state of distributed nodes, but due to its

[3] As we have introduced the new category of synchronicity in the *Communication* aspect (Section V-C), we realigned the *Collaboration type* category accordingly. Labels "real-time" and "off-line" were extracted from the studies.

TABLE III: Support for Collaboration

Collaboration types – The parameters of this category have been renamed.

Value	*Original (2003–2015)*	*Update (2016–2020)*	Δ
Real-time	28 (58%)	19 (66%)	↑
Off-line	24 (50%)	12 (41%)	↓
Prescribed workflow			
Supported	43 (90%)	15 (52%)	↓
Consistency model – New category.			
Eventual	17 (35%)	11 (38%)	↗
Strong	27 (56%)	10 (34%)	↓
Strong eventual	3 (6%)	8 (28%)	↑
Conflict management approach			
Allow & Resolve	27 (56%)	11 (38%)	↓
Preventive	12 (25%)	8 (28%)	↗
Conflict awareness (user) – New category.			
Warning	11 (23%)	21 (72%)	↑
Prompt action	3 (6%)	2 (7%)	↗
Locking – The parameters of this category have been renamed.			
No support	35 (73%)	22 (76%)	↗
Pessimistic	12 (25%)	5 (17%)	↓
Optimistic	1 (2%)	2 (7%)	↗
Conflict resolution type			
(Semi)automated	10 (21%)	13 (45%)	↑
Manual	17 (35%)	8 (28%)	↓
Diff/merge domain – New category.			
Syntactic	46 (96%)	22 (76%)	↓
Semantic	2 (4%)	5 (17%)	↑
Versioning – The parameters of this category have been renamed.			
External: generic VCS	4 (8%)	4 (14%)	↑
Internal	9 (19%)	3 (10%)	↓
External: model-driven	5 (10%)	3 (10%)	→
Network architecture – The parameters of this category have been restructured.			
Centralized single server	40 (83%)	19 (66%)	↓
Centralized multiple srvrs	3 (6%)	6 (21%)	↑
Mixed	3 (6%)	3 (10%)	↗
P2P	2 (4%)	3 (10%)	↗

↑ Increase ↗ Slight increase → No change ↘ Slight decrease ↓ Decrease

Data from original study. Data from update.

underlying mechanisms, it significantly hinders the scalability and user experience of collaborative modeling tools [43]. Eventual consistency provides the weaker guarantee that changes will be eventually observed across each node [44]. Strong eventual consistency combines the benefits of both models and, consequently, it is very suitable for underpinning collaborative applications [45]. We see a prominent decrease in the popularity of strong consistency, changing from 56% to 34%; while the strong eventual model is gaining popularity. Over 28% of approaches support strong eventual consistency, a sharp increase from the previous 6%. For example, [P18] addresses collaborative conflicts on the data level by relying on conflict-free replicated datatypes [45].

Conflicts arise inevitably when several collaborators work in parallel, for instance, during merging of collaborative models. Similar to the benefits of finding problems earlier in the software development process, adopting preventive approaches to *manage conflicts* during collaboration activities saves the effort to be spent in resolving conflicts later in the modeling process. We see a slightly increasing trend (from 25% to 28%) for preventive conflict management approaches in the primary studies. Most of the preventive approaches warn the modelers beforehand (72%), when there are potential chances of conflicts. However, managing those conflicts is left up to the users. Only [P14], [P29] take a prompt action and eliminate the chances for conflict among the modeling artifacts. *Locking* mechanisms also prevent the conflicts during collaboration, where pessimistic locking allows only a single modeler to work on (part of) a model at once and optimistic locking gives modelers the freedom to decide for proceeding with an update at the time of commit [46]. A small number of primary studies supported locking mechanism, with pessimistic approaches (17%) superseding the optimistic one (7%). Though pessimistic approaches eliminate the chances of conflicts completely, optimistic approaches can be a better choice for maintaining efficiency of overall modeling process. The only change we observed since the original study, is the strong decrease in the reliance on pessimistic locks, with a change from 25% to 17%. The overall trend shows that locking-based collaborative techniques are getting less popular. Finally, if the conflicts are allowed, the modeling frameworks need to provide support means to *resolve these conflicts*. Compared to the original study, a good number of primary studies provide (semi-) automated support for resolution of conflicts, with an increase from 21% to 45%. Conflict resolution, and specifically, *diff/merge*, is mainly addressed at the syntactic level (76%). The support for semantic techniques, however, has increased from 4% to 17%. Especially in approaches emphasizing collaboration across disparate domains, semantic techniques (semantic conflict detection, resolution, diff/merge) are crucial. Examples include [P20] and [P24], [P25].

External generic *version control* systems, and especially Git, are the most common among the approaches providing *versioning* support. Approaches such as MetaEdit+ [P13] and MONDO [P06] directly address well-known industrial requirements by this choice. Internal versioning lost popularity, and their support has decreased from 19% to 10%. WebGME [P14] and Collaboro [P10] are examples of such approaches.

The ratio of centralized *network architectures* has slightly decreased. The detailed view of the centralized cases shows that it is the single-server model (e.g., [41], [P29]) that experienced a strong decrease, from 83% to 66%; while the support for centralization by multiple servers (e.g., [P06], [P19], [P26]) increased from 6% to 21%. A slight increase in P2P and mixed (P2P with centralization) architectures can be observed. Such distributed architectures require more intricate data-level consistency considerations, as demonstrated in [P18] (by conflict-free replicated datatypes), and [P08] (by blockchains).

C. Communication

Communication features are responsible for allowing a semantically rich exchange among the involved stakeholders, to augment the information carried by the models they collaborate over. Typical means of communication are chats, wikis, model annotations, comments, change proposals, and forums.

As shown in Table IV, there is a significant shift from synchronous to asynchronous *communication* support. Asynchronous communication has increased from to 33% to 72%,

TABLE IV: Support for Communication

Value	Original (2003–2015)	Update (2016–2020)	∆
Communication type – New category.			
Asynchronous	16 (33%)	21 (72%)	↑
Synchronous	20 (42%)	8 (28%)	↓
Built-in communication means (with at least 3 occurrences)			
Chat	18 (37%)	8 (27%)	↓
Comments	8 (17%)	5 (17%)	→
Call-for-attention	1 (2%)	5 (17%)	↑
Annotations	13 (27%)	3 (10%)	↓
Other	8 (17%)	8 (28%)	↑
Stakeholder types			
Technical	47 (98%)	28 (97%)	↘
Non-technical	7 (15%)	11 (38%)	↑
Workspace awareness score			
High	11 (23%)	11 (38%)	↑
Medium	8 (17%)	5 (17%)	→
Low	29 (60%)	13 (45%)	↓

↑ Increase ↗ Slight increase → No change ↘ Slight decrease ↓ Decrease
Data from original study. Data from update.

as compared to the original study, while synchronous communication, has gotten less supported, strongly decreasing from 42% to 28%. This trend is somewhat counter-intuitive, considering the increased support for real-time collaboration and its good fit for synchronous communication.

The categories of *Client type*, *Collaboration type* and *Communication type* of the respective dimensions of Model management, Collaboration and Communication, are strongly related. Synchronous communication is always supported in real-time collaboration; and appeared more relevant for web- and mobile clients in our primary studies. Only [P16], [P21] provided synchronous communication for desktop clients.

Our primary studies include a variety of both built-in and external communications tools. *Built-in communication* is integrated into the collaborative MDSE approach [8]: chat, comments, call-for-attention and annotations were used more frequently (Table IV). External communication is only prescribed by the collaborative MDSE approach: e-mails and face-to-face discussions were usually preferred for external communication. We observe that, despite being of cardinal importance, the use of external communication tools is either not discussed or only mentioned very implicitly. For instance, [P27] states that the moderator invites the collaborators but does not mention how this invitation is sent. Similarly [P24] does not explain how the document is forwarded (and notified) to next reviewer during sequential collaboration.

Collaboration support for non-technical *stakeholders* has improved. This might be an indication that there is an increasing trend of having tight collaborations between technical (e.g., engineers and developers) and non-technical stakeholders (e.g., business analyst, domain expert). Most approaches are dedicated to specific stakeholders, typically developers, designers, modelers, and business stakeholders. We did not find approaches designed for non-technical stakeholders only; this is in line with the scope of the study, as collaborative MDSE for performing software engineering activities, requires some level of technical proficiency. Interestingly, [P16] included a chatbot as a stakeholder, showing the integration of automated

TABLE V: Recurring limitations

Cluster	Original (2003–2015)	Update (2016–2020)	∆
1. Model management	11	27	↑
2. Evaluation	0	25	↑
3. Collaboration	30	23	↓
4. NFP	13	11	↘
5. Tool improvement	4	5	↗
6. Communication	6	3	↘
Total	63	97	↑

↑ Increase ↗ Slight increase → No change ↘ Slight decrease ↓ Decrease
Data from original study. Data from update.

conversation in collaborative modeling activities.

The original study suggested further research to enhance the support for updating stakeholders about each others' actions in a shared workspace, that is, to increase the *workspace awareness* among stakeholders. It is encouraging to report that we observed an increase in the proportion of studies with a high awareness level, which shows that researchers have taken up this line of research. Especially version control system based mechanisms (e.g., in [P03] and [P06]), and real-time model updates (e.g., in [P22], [P28]) are common.

VI. Challenges and Shortcomings (RQ2)

In this section, we report the results of our update regarding RQ2, i.e., the challenges and shortcomings of collaborative MDSE approaches researchers are facing and have either identified them as actual limitations or suggested addressing the shortcoming as a future work. For each primary study, we have collected the limitations and future works by a thorough analysis of the full text, and applied a card sorting technique [47] to cluster this information.

We have extracted a total number of 97 unique shortcomings from the 29 studies, an average of 3.34 per study. This number is nearly three times as much as the average number of shortcomings identified in the original study (63 limitations in 48 papers, 1.31 average). Plausible explanations to this phenomenon can be the overall better quality of publications in the update (studies appropriately detailing the shortcomings of the approach they describe); or the improved overall maturity of the field (improving publication standards and practices).

Five clusters have emerged from the card sorting, four of them consisting of at least 11 studies, as shown in Table V. Three clusters are identical to the dimensions of collaborative MDSE: model management (with 27 unique limitations identified), collaboration (23), and communication (3); one cluster is related to non-functional properties (11); and one cluster is related to the lack or extensiveness of evaluation (25). The original study identifies the first four clusters. Evaluation-related shortcomings, however, are completely new, and emerged as the second most frequently encountered type.

Model management and collaboration are the most frequently encountered shortcomings related to the three dimensions of collaborative MDSE. The third dimension, communication draws significantly less attention, and shows a decreasing trend. Given the takeaways of Section V-C, we find it likely that communication support, as a dimension of collaborative MDSE, does not draw enough attention in general. Only [P10],

[P16], and [P24] report communication-related shortcomings. Within *model management*, the low number of supported languages (e.g., [P10]), and their expressiveness (e.g., [P23]) is the leading limitation. The lack or restricted support for meta-modeling and multi-view modeling is the second most frequent limitation, e.g., in [P08] and [P28]. As discussed in Section V, the shift towards higher meta-levels has been noticeable during the past five years and, as such, seeing the lack of meta-level support as a limitation might be justified.

In terms of *evaluation*, collaborative approaches have a systemic issue with, in some cases, the complete lack of proper evaluation. It is important to note, that these are self-declared limitations by the authors of primary studies. Typical examples include: evaluation on synthetic or academic examples (e.g., [P05], [P29]), and the size of the case being considered not representative (e.g., [P27]). Another recurring theme is the desire to test the prototype in more realistic settings, often labeled as "industrial" (e.g., [P14], [P15]) or "real" [P28].

Out of the limitations related to *collaboration*, conflict detection and automation of collaboration are the most relevant clusters. Some approaches completely ignore conflict detection and plan it as future work [P01]. More advanced approaches, such as [P07], aim to extend their capabilities with AI-based and semantic conflict detection techniques. As discussed in Section V, semantic techniques are rarely encountered, and thus, this direction allows significant room to grow.

Non-functional properties are a prominent cluster as well. Scalability has been identified as the leading non-functional limitation or desired future work (e.g., [P05], [P11], [P23]). The focus on scalability as a limitation seems to be justified in the case of collaborative tools, potentially supporting the work of distributed, large-scale teams.

The remaining limitations are related to tool development and integration tasks (e.g., [P09], [P23]); and approach-specific concerns, such as supporting crowd modeling in [P26], and supporting abbreviations in the NLP technique of [P22].

VII. PUBLICATION TRENDS (RQ3)

In this section, we report the results of our update regarding RQ3: the publication trends in the domain of collaborative MDSE. For the sake of completeness, we analyze the included publications on an individual basis, rather than in a clustered way. For each primary study we extracted the publication year and type. The results are shown in Figure 2.

During the period of 2016–2020, we have sampled a total of 51 publications. Additionally, for the year 2015, we have sampled 3 publications which could not be included in the original study due to their later publication date. In sum, 2015 has produced 12 publications. This shows a steady publication trend with the average number of 10 publications between 2016–2020. This average is aligned with the previous 5-year period (2011–2015), which produced an average of 9 papers per year. The original study identifies 2003 as the point when the publication trends have started to increase significantly. In comparison with the past five years, the period between 2003–2015 produced an average of 8.15 papers per year. Because of the current update, we were able to identify a significant[4] increase in publication output starting from 2013. This trend can be explained by two components: (i) the appearance of novel, web-based modeling tools that inherently embraced collaborative modeling and remained relevant through multiple years, such as AToMPM [19] and Flexisketch [48] in 2013, and WebGME in 2014 [20]; and (ii) the appearance of collaborative frameworks and platforms with multiple applications, such as the Yjs/Syncmeta [49] framework in 2015, the MONDO platform [50] in 2016, and the SOCIO [51] framework in 2017.

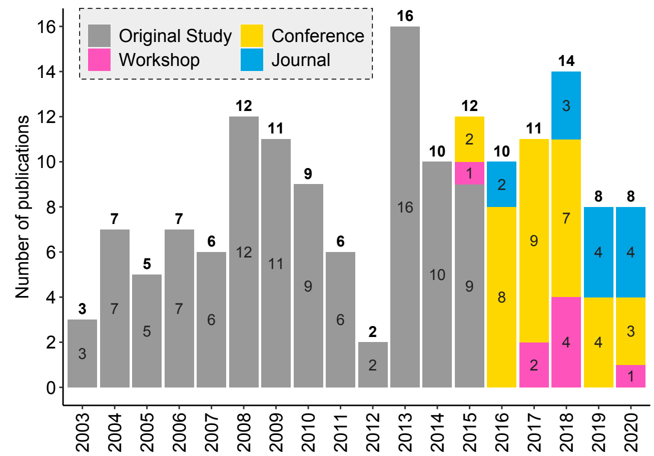

Fig. 2: Publication trends between 2003–2020 with the papers of the current study highlighted.

These figures suggest that the field is relatively steady, with a regular influx of new techniques and approaches. Apart from 2012, which has seen only 2 publications, there were at least 6 papers produced annually in the past 15 years.

TABLE VI: Venues with more than one publication

Venue	Publications	Ratio
MODELS	9	17%
SOSYM, STAF	3 each	11%
ASE, IEEE Software, ER, MODELSWARD, WET-ICE	2 each	18%
Others	29	54%

The ratio of journal and conference papers has increased from 79% in 2011–2015 to 86% in 2016–2020, suggesting the improved maturity of the field. Relevant change can be observed in the venues of publications. Between 2003–2015, the distribution of papers across venues was relatively even, as 8 conferences and workshops have contributed 3–5 papers. The period of 2016–2020 (Table VI) is less even, and we see three venues attracting researchers of the field. 9 papers (17%) were published at the *International Conference on Model Driven Engineering Languages and Systems – MODELS* (6 conference and 3 workshop papers); the journal on *Software & Systems Modeling – SOSYM*, and the *Software Technologies:*

[4]Means until 2012 and from 2013: 6.8 and 11.125, respectively. We used the independent 2-group Mann-Whitney U Test of unequal sample sizes due to the non-parametric nature of the data ($\alpha = 0.05$, $p = 0.014$).

Applications and Foundations – STAF conference contributed 3 papers each.

VIII. Orthogonal findings

We have further analyzed the extracted data to find relevant phenomena orthogonal to the vertical analysis, emerging between specific combinations of concepts in the classification framework. By evaluating the orthogonal findings of the original study with the 2016–2020 data, we have found that each of them are still valid. The reader is referred to the original study (Section 10, pp 22–23) for further information. Therefore, in this section, we only focus on our findings from the 2016–2020 period uncovered by the updated classification framework.

1) Orthogonal findings about Real-time collaboration: Real-time collaboration is naturally coupled with *synchronous communication*. This has been demonstrated by the support for real-time collaboration in synchronous means of communication being 100%, but only 52% in asynchronous ones. In off-line approaches, synchronous and asynchronous means are supported evenly. Real-time communication approaches score higher in *workspace awareness*. 68% of real-time approaches earned a *Medium* or *High* score in this aspect; while this number is 25% in off-line approaches. As a potential limitation, real-time collaboration is frequently supported by a strong notion of model consistency (42%). This is somewhat unexpected as strong eventual (32%) and eventual (26%) consistency models fit the real-time paradigm better [44].

2) Orthogonal findings about Mobile clients: While mobile clients are naturally restricted in their resources and physical dimensions, they evidently align well with supporting non-technical stakeholders in a collaborative modeling endeavor. We have observed each approach with a mobile client supports both technical technical and non-technical users. The support for non-technical users is much lower in desktop-based (33%) and browser-based (43%) approaches. Approaches with mobile clients also tend to be more lightweight, where the real-time interaction is important (100% support), but version control and conflict awareness are omitted.

3) Orthogonal findings about Conflict management and Conflict resolution: Preventive conflict management is most frequently encountered in single-view settings (88%); only one preventive approach was found in multi-view settings (12%) [P24]. 88% of preventive techniques are found in real-time settings; allow-and-resolve techniques, however, distribute across real-time and off-line settings evenly. The automation of conflict resolution is a stronger trend in single-view settings than multi-view ones. 69% of single-view settings provide some means of automation for resolving conflicts; while this number is 50% in multi-view settings. This is likely because of the more intricate nature of conflicts in multi-view settings. While conflicts in single-view settings are purely syntactic, conflicts in multi-view settings tend to be more semantic, and thus, harder to detect and repair.

4) Orthogonal findings about the Communication type: There is a noticeable difference in how much different client types support different communication types. Asynchronous and synchronous means of communication are supported nearly evenly in desktop clients (100% and 87% respectively). This ratio, however substantially different in web and mobile clients. In web-based clients, synchronous communication is over 1.5 times more often found than asynchronous (100% and 64% support respectively). In mobile clients this ratio is even higher (100% and 25% support respectively). Multi-view approaches distribute evenly across synchronous and asynchronous communication types. Synchronous communication is typically supported by chat, comments, calls for attention and annotations; while asynchronous communication is typically supported by proposals, reviews, annotations and comments. While allow-and-resolve conflict management techniques are equally frequent with both types of communication, preventive conflict management techniques are nearly twice as frequent in synchronous than in asynchronous communication (7 and 4 respectively).

IX. Discussion

In this section, we summarize the takeaways, identify future directions to improve the quality and maturity of the field, reflect on the methodology, and discuss the threats to validity.

A. Key takeaways and recommendations

We observed a strong imbalance among the three dimensions of collaborative MDSE, with **communication being severely overlooked**, both in terms of supported communication techniques and in terms of planned future work. Better integrated means of communication, preferably of the synchronous type, are required to seamlessly augment the modeling process with the ability to interact in natural language. We recommend treating the current lack of communication facilities with a special emphasis in the next generation of collaborative modeling tools. A case for model-augmented chat facilities has been made in the context of chatbots [P22]. We foresee similar mechanisms appearing in human-human interactions. We suggest researching the possibilities of making model elements first-class citizens in chats, wikis, and similar settings, possibly integrated with natural language processing to further elevate the quality of human-computer interaction.

Relaxed consistency models have become more prominent. Especially eventual and strong eventual models have gained traction. We welcome this trend as these models are especially suitable for developing modeling tools with an elevated user experience. Approaches, such as blended modeling [52], relaxed design, and prototyping, demand more freedom in temporarily deviating from well-formedness, correctness and consistency, and thus, rely on proper inconsistency tolerance mechanisms. Model **consistency has become a first-class concept at the level of physical data** too, in the form of conflict-free replicated data types (CRDT), motivated by use-cases of real-time collaboration. Such approaches align well with cross-domain settings where common modeling concepts cannot be assumed, and thus, physical data types might be the prime candidates to carry consistency information or prevent conflicts by design. We suggest the research on such relaxed consistency models to continue. Furthermore, we see opportunities in developing advanced consistency models explicitly targeting heterogeneous modeling settings where common abstract syntaxes and metamodels cannot be assumed, and semantic reasoning might be required [29].

Mobilization of modeling is trending, as the ratio of mobile and browser clients has been increasing. This new generation of modeling tools, however, still mainly relies on strong consistency models, e.g., AToMPM [P05] and Collaboro [P10]. On a related note, **real-time collaboration is becoming increasingly popular**, and it is currently the preferred option over off-line techniques. The split in the support between real-time and off-line collaboration has increased from 8.33% to nearly 25%. We recommend tools builders to incorporate state-of-the-art consistency models (e.g., strong eventual consistency [45]) and inconsistency management techniques, as this choice has a profound impact on the eventual usability and performance of collaborative modeling tools.

The **lack of systematic evaluation frameworks** poses a serious issue, as it hinders the applicability of academic results. This is a glaring need to be addressed, especially considering that the maturity of the field now attracts tool builders outside of academia [11]–[13]. Systematic evaluation frameworks will pave the road for conducting (replicable and independently verifiable) empirical studies on collaborative modeling techniques, algorithms, heuristics, and tools, thus making this research field scientifically solid and robust. In the interim, we urge the community to evaluate collaborative MDSE tools in a hands-on fashion, e.g., via workshops and tool challenges, such as the HoWCoM workshop at this year's MODELS conference [10].

On a positive note, the **maturity** of the field of collaborative MDSE has improved, demonstrated by steady publication trends and improved quality of publications. With 51 studies published in the last five years, the amount of papers per year has doubled since the original study. The ratio of journal papers has improved, and focused workshops [9], [10] have appeared. We recommend maintaining these good practices. We suggest coordination and collaboration with adjacent communities, such as Human-Computer Interaction, and Human Factors in Modeling [53], e.g. by inviting guest editors to special editions of journals, and inviting keynote speakers to scientific events from these communities.

B. Methodological reflections

Results indicate this update was needed. However, at the time of deciding to carry out this study, we could only rely on the assessment framework discussed in Section III. In retrospect, we find this framework appropriate, and well-designed to aid such decisions. Carrying out a fully recursive forward snowballing, instead of just one iteration suggested by [14], turned out to be a good decision, as 26% of eventually included papers were identified in the second iteration, and additional 2% were identified in the third. This ratio is likely to increase with the frequency of research output of the field (the faster studies are produced, the more studies will appear in later iterations of the snowballing), and with sufficiently long delays between the original study and the update. Therefore, we suggest following a fully recursive approach for future systematic updates in disciplines with a research output of high frequency, such as the software engineering.

C. Threats to validity

We conducted the research reported in this paper based on the carefully designed protocol of the original study [16]. Our work has achieved a 54.5% result in the quality checklist defined by Petersen et al. for systematic studies [27]. This quality score is significantly better than the median and absolute maximum scores (33% and 48%, respectively) reported by [27]. The most important threat to the external validity is the search method that was limited to a forward snowballing, as suggested by [14]. In order to mitigate this threat, we have carried out a fully recursive forward snowballing. In terms of internal validity, the most important threat was introduced in our preliminary assessment of the need for an update. When assessing the volume of potentially relevant new literature (Section III-B), we mitigated the threat by measuring the ratio of volumes between the target period and baseline period in a uniform way: by running the same search string on Google Scholar for both periods. A slight threat to validity remained due to the false positive inflation problem of Google Scholar [14]; but eventually, this did not influence the results. A slight threat to the conclusion validity was introduced in Section VII, when carrying out a significance test over relatively small samples of annual research output. We have validated the plausibility of the conclusion by identifying the underlying trends in technical contributions.

X. Conclusions and Future Work

In this paper, we have presented the results of our systematic update on the mapping study by Franzago et al. [8] on collaborative MDSE in the period of 2016–2020. Starting from over 880 studies, we have selected 54 primary and one secondary studies through a rigorous process. We extended the classification framework of the original study based on the newly identified literature, and extracted insightful information. We have identified multiple interesting trends, and based on these, we have outlined the important research directions of the field. As a next step, we will conduct the non-academic counterpart of this study, focusing on the state of the practice and practitioners' needs so as to identify relevant gaps and opportunities between academia and industry. We suggest a focused work on the communication dimension of collaborative MDSE, which proved to be by far the most underdeveloped and overlooked dimension of the three. Finally, we suggest revisiting this study again in five years, and assessing the need for another update, based on our adapted protocol.

Acknowledgements

This research was partially funded by the Rijksdienst voor Ondernemend Nederland (RVO) through the ITEA3 BUMBLE project (18006). The authors would like to thank Théo Le Calvar for his constructive remarks.

PRIMARY STUDIES

[P01] M. Appeldorn, D. Kuryazov, and A. Winter, "Delta-driven collaborative modeling." in *MODELS Workshops*, 2018, pp. 293–302.

[P02] F. B. Aydemir and F. Dalpiaz, "Supporting Collaborative Modeling via Natural Language Processing," in *International Conference on Conceptual Modeling*. Springer, 2020, pp. 223–238.

[P03] J. Y. Bang, Y. Brun, and N. Medvidovic, "Continuous Analysis of Collaborative Design," in *2017 IEEE International Conference on Software Architecture (ICSA)*. IEEE, 2017, pp. 97–106.

[P04] S. Bennani, I. A. E. Kouch, M. E. Hamlaoui, S. Ebersold, B. Coulette, and M. Nassar, "A Formalization of Group Decision Making in Multi-Viewpoints Design," *arXiv preprint arXiv:2004.14098*, 2020.

[P05] J. Corley, E. Syriani, and H. Ergin, "Evaluating the cloud architecture of AToMPM," in *2016 4th International Conference on Model-Driven Engineering and Software Development (MODELSWARD)*. IEEE, 2016, pp. 339–346.

[P06] A. Gómez, X. Mendialdua, K. Barmpis, G. Bergmann, J. Cabot, X. de Carlos, C. Debreceni, A. Garmendia, D. S. Kolovos, and J. de Lara, "Scalable modeling technologies in the wild: an experience report on wind turbines control applications development," *Software and Systems Modeling*, pp. 1–33, 2020.

[P07] L. Halilaj, I. Grangel-González, M.-E. Vidal, S. Lohmann, and S. Auer, "SerVCS: Serialization Agnostic Ontology Development in Distributed Settings," in *International Joint Conf. on Knowledge Discovery, Knowledge Engineering, and Knowledge Management*. Springer, 2016, pp. 213–232.

[P08] F. Härer, "Decentralized Business Process Modeling and Instance Tracking Secured by a Blockchain," 2018.

[P09] B. Horváth, Á. Horváth, and M. Wimmer, "Towards the Next Generation of Reactive Model Transformations on Low-Code Platforms: Three Research Lines," in *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, 2020, pp. 1–10.

[P10] J. L. C. Izquierdo and J. Cabot, "Collaboro: a collaborative (meta) modeling tool," *PeerJ Computer Science*, vol. 2, p. e84, 2016.

[P11] Y. Jiang, S. Wang, K. Fu, W. Zhang, and H. Zhao, "A collaborative conceptual modeling tool based on stigmergy mechanism," in *Proceedings of the 8th Asia-Pacific Symposium on Internetware*, 2016, pp. 11–18.

[P12] N. Kanagasabai, O. Alam, and J. Kienzle, "Towards Online Collaborative Multi-View Modelling," in *International Conference on System Analysis and Modeling*. Springer, 2018, pp. 202–218.

[P13] S. Kelly, "Collaborative Modelling with Version Control," in *Federation of International Conferences on Software Technologies: Applications and Foundations*. Springer, 2017, pp. 20–29.

[P14] T. Ma and J. Sallai, "MiW: A domain specific modeling environment for complex molecular systems," *Procedia Computer Science*, vol. 108, pp. 1232–1241, 2017.

[P15] M. Manca, F. Paternò, and C. Santoro, "Collaborative Task Modelling on the Web," in *Human-Centered and Error-Resilient Systems Development*. Springer, 2016, pp. 317–334.

[P16] N. A. Moketar, M. Kamalrudin, S. Sidek, M. Robinson, and J. Grundy, "An automated collaborative requirements engineering tool for better validation of requirements," in *Proceedings of the 31st IEEE/ACM Intl. Conf. on Automated Software Engineering*, 2016, pp. 864–869.

[P17] R. A. C. Neto, K. Farias, V. Bischoff, and L. J. Gonçales, "Supporting collaborative modelling in UML class diagrams," *Revista Brasileira de Computação Aplicada*, vol. 11, no. 3, pp. 39–46, 2019.

[P18] P. Nicolaescu, M. Rosenstengel, M. Derntl, R. Klamma, and M. Jarke, "Near real-time collaborative modeling for view-based web information systems engineering," *Information Systems*, vol. 74, pp. 23–39, 2018.

[P19] J. Oppenlaender, T. Tiropanis, and S. Hosio, "CrowdUI: Supporting Web Design with the Crowd," *Proceedings of the ACM on Human-Computer Interaction*, vol. 4, no. EICS, pp. 1–28, 2020.

[P20] S. Oppl, "Articulation of work process models for organizational alignment and informed information system design," *Information & Management*, vol. 53, no. 5, pp. 591–608, 2016.

[P21] J. L. Pérez-Medina and J. Vanderdonckt, "Sketching by Cross-Surface Collaboration," in *International Conference on Information Technology & Systems*. Springer, 2019, pp. 386–397.

[P22] S. Pérez-Soler, E. Guerra, and J. de Lara, "Collaborative Modeling and Group Decision Making Using Chatbots in Social Networks," *IEEE Software*, vol. 35, no. 6, pp. 48–54, 2018.

[P23] G. Sunyé, "Model Consistency for Distributed Collaborative Modeling," in *European Conference on Modelling Foundations and Applications*. Springer, 2017, pp. 197–212.

[P24] M. A. Tröls, A. Mashkoor, and A. Egyed, "Multifaceted Consistency Checking of Collaborative Engineering Artifacts," in *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*. IEEE, 2019, pp. 278–287.

[P25] Y. Van Tendeloo and H. Vangheluwe, "Unifying Model- and Screen Sharing," in *2018 IEEE 27th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. IEEE, 2018, pp. 127–132.

[P26] D. Vaquero-Melchor, J. Palomares, E. Guerra, and J. de Lara, "Active domain-specific languages: Making every mobile user a modeller," in *2017 ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems (MODELS)*. IEEE, 2017, pp. 75–82.

[P27] T. T. Vo, B. Coulette, H. N. Tran, and R. Lbath, "An Approach to Define and Apply Collaboration Process Patterns for Software Development," in *International Conference on Model-Driven Engineering and Software Development*. Springer, 2015, pp. 248–262.

[P28] D. Wüest, N. Seyff, and M. Glinz, "Flexisketch: a lightweight sketching and metamodeling approach for end-users," *Software & Systems Modeling*, vol. 18, no. 2, pp. 1513–1541, 2019.

[P29] M. Zerrouk, A. Anwar, and I. Benelallam, "Managing Model Conflicts in Collaborative Modeling Using Constraint Programming," in *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*. IEEE, 2018, pp. 117–123.

REFERENCES

[1] A. Bucchiarone, F. Ciccozzi, L. Lambers, A. Pierantonio, M. Tichy, M. Tisi, A. Wortmann, and V. Zaytsev, "What is the future of modeling?" *IEEE Software*, vol. 38, no. 2, pp. 119–127, 2021.

[2] D. S. Kolovos, L. M. Rose, N. Matragkas, R. F. Paige, E. Guerra, J. S. Cuadrado, J. De Lara, I. Ráth, D. Varró, M. Tisi *et al.*, "A research roadmap towards achieving scalability in model driven engineering," in *Proceedings of the Workshop on Scalability in Model Driven Engineering*, 2013, pp. 1–10.

[3] M. Brambilla, J. Cabot, and M. Wimmer, "Model-driven software engineering in practice," *Synthesis lectures on software engineering*, vol. 3, no. 1, pp. 1–207, 2017.

[4] H. Muccini, J. Bosch, and A. van der Hoek, "Collaborative modeling in software engineering," *IEEE Software*, vol. 35, no. 6, pp. 20–24, 2018.

[5] Y. Zhang and S. Patel, "Agile model-driven development in practice," *IEEE software*, vol. 28, no. 2, pp. 84–91, 2010.

[6] R. Sanchis, Ó. García-Perales, F. Fraile, and R. Poler, "Low-code as enabler of digital transformation in manufacturing industry," *Applied Sciences*, vol. 10, no. 1, p. 12, 2020.

[7] J. Cabot, "Positioning of the low-code movement within the field of model-driven engineering," in *Model Driven Engineering Languages and Systems: Companion Proceedings*, 2020, pp. 1–3.

[8] M. Franzago, D. Di Ruscio, I. Malavolta, and H. Muccini, "Collaborative model-driven software engineering: a classification framework and a research map," *IEEE Transactions on Software Engineering*, vol. 44, no. 12, pp. 1146–1175, 2017.

[9] "International Workshop on Collaborative Modelling in MDE," https://cs.gssi.it/commitmde2016/, Retrieved: 28/04/2021.

[10] "Hands-on Workshop on Collaborative Modeling," http://howcom2021.github.io/, Retrieved: 11/05/2021.

[11] "MONDO Project official webpage," http://www.mondo-project.org/, Retrieved: 28/04/2021.

[12] "BUMBLE Project official webpage," https://itea3.org/project/bumble.html, Retrieved: 28/04/2021.

[13] "Lowcomote official webpage," https://www.lowcomote.eu, Retrieved: 28/04/2021.

[14] E. Mendes, C. Wohlin, K. Felizardo, and M. Kalinowski, "When to update systematic literature reviews in software engineering," *Journal of Systems and Software*, vol. 167, p. 110607, 2020.

[15] C. Masson, J. Corley, and E. Syriani, "Feature model for collaborative modeling environments." in *MODELS*, 2017, pp. 164–173.

[16] M. Franzago, D. Di Ruscio, I. Malavolta, and H. Muccini, "Protocol for a Systematic Mapping Study on Collaborative Model-Driven Software Engineering," Tech. Rep., 2016, Retrieved: 28/04/2021. [Online]. Available: http://arxiv.org/abs/1611.02619

[17] M. Stephan, "Emerging concepts and trends in collaborative modeling: A survey." in *MODELSWARD*, 2019, pp. 240–247.

[18] A. M. Ertugrul and O. Demirors, "An exploratory study on role-based collaborative business process modeling approaches," in *Proceedings of the 7th International Conference on Subject-Oriented Business Process Management*, 2015, pp. 1–5.

[19] E. Syriani, H. Vangheluwe, R. Mannadiar, C. Hansen, S. Van Mierlo, and H. Ergin, "AToMPM: A Web-based Modeling Environment," *Demos/Posters/StudentResearch - MoDELS*, vol. 2013, pp. 21–25, 2013.

[20] M. Maróti, T. Kecskés, R. Kereskényi, B. Broll, P. Völgyesi, L. Jurácz, T. Levendovszky, and Á. Lédeczi, "Next generation (meta) modeling: web-and cloud-based collaborative tool infrastructure," *MPM-MoDELS*, vol. 1237, pp. 41–60, 2014.

[21] M. Dirix, A. Muller, and V. Aranega, "Genmymodel: an online uml case tool," in *ECOOP*, 2013.

[22] "Obeo Designer Team," https://www.obeodesigner.com/en/collaborative-features, Retrieved: 11/05/2021.

[23] A. Bucchiarone, J. Cabot, R. F. Paige, and A. Pierantonio, "Grand challenges in model-driven engineering: an analysis of the state of the research," *Software and Systems Modeling*, vol. 19, no. 1, pp. 5–13, 2020.

[24] V. Garousi and J. M. Fernandes, "Highly-cited papers in software engineering: The top-100," *Information and Software Technology*, vol. 71, pp. 108–128, 2016.

[25] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," Technical report, Ver. 2.3 EBSE Technical Report. EBSE, Tech. Rep., 2007.

[26] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer Science & Business Media, 2012.

[27] K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," *Information and Software Technology*, vol. 64, pp. 1–18, 2015.

[28] C. Debreceni, G. Bergmann, I. Ráth, and V. Dániel, "Property-based locking in collaborative modeling," in *2017 ACM/IEEE 20th International Conference on Model Driven Engineering Languages and Systems (MODELS)*. IEEE, 2017, pp. 199–209.

[29] I. David, J. Denil, K. Gadeyne, and H. Vangheluwe, "Engineering process transformation to manage (in) consistency," in *1st International Workshop on Collaborative Modelling in MDE, COMMitMDE 2016*, 2016, pp. 7–16.

[30] C. Wohlin, E. Mendes, K. R. Felizardo, and M. Kalinowski, "Guidelines for the search strategy to update systematic literature reviews in software engineering," *Information and Software Technology*, vol. 127, p. 106366, 2020.

[31] P. Garner, S. Hopewell, J. Chandler, H. MacLehose, E. A. Akl, J. Beyene, S. Chang, R. Churchill, K. Dearness, G. Guyatt *et al.*, "When and how to update systematic reviews: consensus and checklist," *bmj*, vol. 354, 2016.

[32] I. David, K. Aslam, S. Faridmoayer, I. Malavolta, E. Syriani, and P. Lago, "Collaborative MDSE MODELS 2021 Replication Package," https://zenodo.org/record/5128416, 2021.

[33] V. R. Basili, G. Caldiera, and H. D. Rombach, "The Goal Question Metric Approach," in *Encyclopedia of Software Engineering*. Wiley, 1994, vol. 2, pp. 528–532.

[34] M. Gusenbauer, "Google scholar to overshadow them all? comparing the sizes of 12 academic search engines and bibliographic databases," *Scientometrics*, vol. 118, no. 1, pp. 177–214, 2019.

[35] B. Bigendako and E. Syriani, "Modeling a tool for conducting systematic reviews iteratively," in *Model-Driven Engineering and Software Development*, vol. 1. Scitepress, 2018, pp. 552–559.

[36] H. Zhang, M. A. Babar, and P. Tell, "Identifying relevant studies in software engineering," *Information and Software Technology*, vol. 53, no. 6, pp. 625–637, 2011.

[37] M. Vaismoradi, H. Turunen, and T. Bondas, "Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study," *Nursing & health sciences*, vol. 15, no. 3, pp. 398–405, 2013.

[38] D. S. Cruzes and T. Dybå, "Research synthesis in software engineering: A tertiary study," *Information and Software Technology*, vol. 53, no. 5, pp. 440–455, 2011.

[39] A. G. Kleppe, J. Warmer, J. B. Warmer, and W. Bast, *MDA explained: the model driven architecture: practice and promise*. Addison-Wesley Professional, 2003.

[40] D. Steinberg, F. Budinsky, E. Merks, and M. Paternostro, *EMF: eclipse modeling framework*. Pearson Education, 2008.

[41] J. Corley, E. Syriani, H. Ergin, and S. Van Mierlo, "Cloud-based multi-view modeling environments," in *Modern Software Engineering Methodologies for Mobile and Cloud Environments*. IGI Global, 2016, pp. 120–139.

[42] A. Cicchetti, F. Ciccozzi, and A. Pierantonio, "Multi-view approaches for software and system modelling: a systematic literature review," *Software and Systems Modeling*, vol. 18, no. 6, pp. 3207–3233, 2019.

[43] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," in *Concurrency: the Works of Leslie Lamport*, 2019, pp. 179–196.

[44] W. Vogels, "Eventually consistent," *Communications of the ACM*, vol. 52, no. 1, pp. 40–44, 2009.

[45] M. Shapiro, N. Preguiça, C. Baquero, and M. Zawirski, "Conflict-free replicated data types," in *Symposium on Self-Stabilizing Systems*. Springer, 2011, pp. 386–400.

[46] K. Altmanninger, M. Seidl, and M. Wimmer, "A survey on model versioning approaches," *International Journal of Web Information Systems(IJWIS)*, vol. 5, no. 3, pp. 271–304, 2009.

[47] D. Spencer, *Card sorting: Designing usable categories*. Rosenfeld Media, 2009.

[48] D. Wüest, N. Seyff, and M. Glinz, "Flexisketch: A mobile sketching tool for software modeling," in *International Conference on Mobile Computing, Applications, and Services*. Springer, 2012, pp. 225–244.

[49] M. Derntl, P. Nicolaescu, S. Erdtmann, R. Klamma, and M. Jarke, "Near real-time collaborative conceptual modeling on the web," in *International Conference on Conceptual Modeling*. Springer, 2015, pp. 344–357.

[50] G. Bergmann, C. Debreceni, I. Ráth, and D. Varró, "Query-based access control for secure collaborative modeling using bidirectional transformations," in *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*, 2016, pp. 351–361.

[51] S. Pérez-Soler, E. Guerra, and J. de Lara, "Assisted modelling over social networks with socio." in *MODELS (Satellite Events)*, 2017, pp. 561–565.

[52] F. Ciccozzi, M. Tichy, H. Vangheluwe, and D. Weyns, "Blended modelling - what, why and how," in *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, 2019, pp. 425–430.

[53] "International Workshop on Human Factors in Modeling / Modeling of Human Factors," https://www.monash.edu/it/humanise-lab/hufamo21, Retrieved: 23/07/2021.