Taking Goal Models Downstream: A Systematic Roadmap

Jennifer Horkoff, Tong Li, Feng-Lin Li, Mattia Salnitri,
Evellin Cardoso, Paolo Giorgini, John Mylopoulos
Department of Information Engineering and Computer Science,
University of Trento, Italy
{horkoff,tong.li,salnitri,paolo.giorgini}@disi.unitn.it
{fenglin.li,evellin.souzacardoso}@unitn.it

João Pimentel
Universidade Federal de Pernambuco, Brazil
jhcp@cin.ufpe.br

***Abstract*—Creating and reasoning with goal models is useful for capturing, understanding, and communicating about requirements in the early stages of information system (re)development. However, the utility of goal models is greatly enhanced when an awareness of system intentions can feed into other stages in the requirements analysis process (e.g. requirements elaboration, validation, planning), and can be used as part of the entire system life cycle (e.g., architecture, process design, coding, testing, monitoring, adaptation, and evolution). In order to understand the progress that has been made in integrating goal models with downstream system development, we ask: what approaches exist which map/integrate/transform goal-oriented languages to other software artifacts or languages? To answer this question, we conduct a systematic survey, producing a roadmap of work summarizing 174 publications. Results include a categorization of the "why?" and "how?" for each approach. Findings show that there are a wide variety of proposals with many proposed sources and targets, covering multiple paradigms, motivated by a variety of purposes. We conclude that although much work has been done in this area, the work is fragmented and is often still in a proposal stage.**

Keywords-requirements engineering; goal model; model transformation; systematic literature survey; systematic literature map; evidence-based requirements engineering

I. Introduction

In order for an information system to be successful, it must adequately address the needs of system stakeholders. Goal-Oriented Modeling Languages (Goal Models) aim to ensure as much, explicitly capturing stakeholder goals, refinements, tradeoffs, alternatives, and responsibilities. Over the last two decades, goal modeling has received much focus as part of requirements and software engineering (RE and SE), but also in the fields of Information Systems, Conceptual Modeling, and Enterprise Modeling. Goal models have been used as an effective means to capture the interactions and information-related requirements of complex information systems [1].

Although the process of creating and reasoning with goal models can be useful for capturing, understanding, and communicating about requirements in the early stages of system (re)development, the utility of goal models is greatly enhanced when model contents can be used as part of downstream development. If goal model concepts are mapped or transformed to downstream system artifacts, such traceability can be used to ensure that system development and operation meets the goals identified as part of early requirements analysis. More specifically, goal model contents should feed into or influence other RE efforts (e.g. requirements elaboration, verification, planning), and should be further used as part of the entire system life cycle, including architecture, process design, code development, testing, monitoring, adaptation, and system evolution. This integration is challenging due to the qualitative, social nature of goal models – it is difficult to take "fuzzy" concepts such as softgoals, roles, and dependencies and map or transform them to concrete functional system elements.

Much existing work has addressed dimensions of this integration problem. Specifically, approaches have provided ways to map, transform or integrate goal models (in)to other RE or SE artifacts. In this work, we aim to understand the landscape of such existing work, evaluating the progress and maturity of efforts in this area. We want to understand the nature of existing proposals for goal model integration, including the type of transformations proposed, the type of goal models used, the motivations for such techniques, the common targets of the transformations, the venues in which this work can be found, the network of paper authors, and the trends in such approaches.

In this paper, we provide an initial roadmap[1] *of approaches which map, transform or integrate goal-oriented languages to or from other artifacts or models related to the software or system lifecyle.*

Work by Kitchenam et al. has advocated for Evidence-based Software Engineering, inspired by Evidence-based Medicine, finding and assessing available evidence to address questions raised as part of software engineering research and practice [22]. In this study, we performed Evidence-based Requirements Engineering (EBRE), finding and summarizing available publications in order to answer goal model-related research questions. Specifically, we have produced a roadmap summarizing publications falling under our scope, without evaluating the relative quality of the work [24]. We place

[1]The type of study we perform is often called a systematic literature map; however, as the subject of our map includes mappings between models, we use the term *roadmap* to describe our study.

particular emphasis on publications classified under Software or Requirements Engineering. The roadmap can be beneficial for several types of readers:

- For researchers interested in using goal-orientated system development, the overview provided by this paper will better enable building upon existing work, avoiding 'reinventing the wheel', helping to understand trends, and guide effort towards new areas.
- From a practitioner perspective, this survey will help demonstrate the ways in which goal-oriented approaches can be integrated into existing system development approaches, providing ideas on how goal-orientation can be adopted in practice, including pointers to work containing further details.

The paper is organized as follows. Sec. II describes the scope of our mapping, while Sec. III describes the research questions and methodology followed. Sec. IV summarizes and analyzes mapping results. Sec. V lists threats to the validity of our study, describing future work which may mitigate these threats. Sec. VI reviews related approaches, while Sec. VII provides conclusions, discussion, and sketches more future work.

II. Scope and Preliminaries

In this section, we make the boundaries of our investigation clear, providing definitions of key survey concepts, used to define the scope of our roadmap. Definitions for Language and Goal-Oriented Language (GOL) are provided in the first two rows of Table I.

In this work we focus on techniques which introduce a transformation, mapping, or integration to or from a GOL to a system-related artifact. In order to allow our roadmap to cover a variety of approaches, we deliberately leave the definition of system-related artifacts broad. Such artifacts can include, for example, models, conceptual artifacts (features, services, agents), and processes.

In order to be included in our map, one source/target of the transformation/mapping/integration must be a GOL. We focus on GOLs which include a structured language, either textual or graphical, formally or informally defined.

We provide definitions for model transformations, mappings, and integrations in Table I, including types of transformations (exogenous/endogenous, horizontal/vertical). When classifying the transformation types of publications, we capture the classifications used by the publication authors whenever possible (e.g. the authors describe their approach as "transforming", "mapping", "integrating") and otherwise make judgments based on the Table I definitions.

As we are interested in transformations and integrations, moving from one phase or view in the system lifecycle to another, we exclude from our study model extensions, where additional concepts are added to an existing language. In our view, model extensions do not make sufficient transitions from one phase or view of the system to another, while integrations are more likely to bridge across conceptual spaces. We differentiate extension from integration by specifying that integrations must integrate two different existing languages. For example, adding a security concept to $i*$ is an extension, and is excluded (e.g., [16]), while combining $i*$ and problems frames is an integration, and is included (e.g., [27]).

We include only those papers published as part of an international journal, conference, symposium or book. We omit workshop papers, local events, and theses, as the peer review process for these types of publications are typically less rigorous. Future work could expand our scope to include these papers, looking especially at highly cited work. Our scoping criteria is summarized in Table II.

We can illustrate our scoping rules using example goal-oriented approaches. An example exogenous vertical transformation would be a transformation from goal models to class diagrams (e.g., [2]). An exogenous horizontal transformation may include a method which transforms a goal model to another high-level requirements modeling language, such as UML use cases (e.g., [13]). An endogenous vertical transformation may include methods which transform a goal model into another goal model with a lower level of abstraction, for example, from requirements to architecture (e.g., [7]). Endogenous horizontal transformation would include, for example, goal model visualization techniques (views, slices), reasoning approaches, refactoring, or syntactical analysis (e.g. [32]). As our focus is on the link from goal models to the system lifecyle we omit approaches which perform only endogenous horizontal transformations. Such approaches are typically aimed to improve use of goal models as part of only the RE stage of system development.

Other transformation classifications, such as syntactical vs. semantical or different technical spaces [28], do not act as inclusion or exclusion criteria in our survey. We are interested in both automatic and manual, bidirectional and unidirectional transformations, as per [10].

III. Survey Methodology

As per Petersen et al. [29], we articulate the specific research questions (RQs) guiding our study. We can identify an overarching research question (**RQ0**), namely: What approaches exist which map/integrate/transform goal-oriented languages to/from other RE/SE software artifacts or languages? Once we have identified approaches, using our scoping criteria from Sec. II, we ask further, more detailed questions, as listed in Table III.

Our process for finding and including or excluding papers is adapted from the processes presented in [29], summarized in Fig. 1. To increase our coverage, we searched for relevant papers by conducting both a systematic search of available research paper databases and by "snowballing", starting with a set of core papers believed to be in-scope, and expanding our set of consideration based on papers referenced by these papers.

Snowballing. We started with a set of 99 core papers found as part of the authors' previous work, and believed to be included by our criteria. Candidate papers were assessed by reading the title and abstract. The reader could optionally look

TABLE I: DEFINITIONS USED TO DEFINE STUDY SCOPE

Language	"A **Language** consists of a syntactic notation (syntax), which is a possibly infinite set of legal elements, together with the meaning of those elements, which is expressed by relating the syntax to a semantic domain. ... Depending on the language type, syntactic elements can be words, sentences, statements, boxes, diagrams, terms, models, clauses, modules, and so on" [17]. Languages can be graphical or textual, and the semantics (meaning) can be formally or informally defined.
Goal-Oriented Language (GOL)	A language which includes the concept of goal as a first class object. Goal-oriented Languages are often graphical (i.e. are modeling languages), having a visual syntax (e.g. Tropos [8], i* [34], KAOS [11], NFR [9], GRL [18], etc.) but may also be textual (e.g., GBRAM [4]).
Transformation	A process that takes one or more source models as input and produces one or more target models as output by following a set of transformation rules [25], [28].
Mapping	A set of rules that describes how one or more constructs in the source modeling language can be connected to one or more constructs in the target modeling language [25], [28].
Integration	The creation of a new modeling language which is made up of constructs and relations from the source and target modeling languages.
Exogenous Transformation	A transformation between models expressed in different languages [28].
Endogenous Transformation	A transformation between models expressed in the same language [28].
Vertical Transformation	A transformation where the source and target models reside at different abstraction levels [28].
Horizontal Transformation	A transformation where the source and target models reside at the same abstraction level [28].

TABLE II: PUBLICATION INCLUSION AND EXCLUSION CRITERIA

Inclusion Criteria	**Exclusion Criteria**
Transforms (maps/integrates) to/from/with a GOL to/from/with an RE or SE artifact or model, and	Describes only model extensions, or
Describes exogenous vertical or horizontal, or endogenous vertical transformations, and	Describes only endogenous horizontal transformations, or
If GOL is formalized, uses formalisms as part of downstream development, and	Formalizes a GOL without using formalisms as part of downstream development, or
In conference, journal, or in/is a book.	In workshops, regional conferences, and theses.

TABLE III: RESEARCH QUESTIONS

RQ1	What types of transformations are used ([mapping/transformation/integration], [horizontal/vertical], [endogenous/exogenous])?
RQ2	What goal modeling frameworks are used most frequently?
RQ3	What sources or targets are goal models mapping/transformed/integration to/from/with? Are there trends in these choices?
RQ4	What are the motivations for the approaches? Are there trends in these motivations?
RQ5	What type of research papers focus on these approaches (validation/evaluation/solution/philosophical/opinion/experience as per [33])?
RQ6	In what journals or conferences do approaches typically appear?
RQ7	What techniques are most widely cited? Are citations equally distributed?
RQ8	Who are the main contributors? What does the network of authors look like?
RQ9	Is interest in goal model integration increasing or decreasing?

at details in the paper, reading the introduction or scanning the paper looking at figures and sections titles. A decision was made to include or exclude the papers based on our criteria (Table II). Further papers were found by looking through the references of core papers, looking for candidate papers based on the paper title and publication venue. In order to ensure the snowballing process ended, we limited our reference search to a depth of two. Future efforts could extend this limit. Overall, at the time of paper submission, we have considered 113 papers in the snowballing process, including 61 papers in our survey.

Systematic Search. In addition, we performed a systematic literature search, searching for publications in several research databases (IEEE, Springer, ACM), published in the last 10 years (2003-2013). We derived our search string from our scope and research questions, searching for:

```
("requirements engineering" OR "software
engineering") AND ("goal model") AND
(transformation OR mapping OR derivation OR
alignment OR integration OR link)
```

where we replaced "`goal model`" with a variety of common forms (e.g., "goal modeling", "goal-oriented requirements").

The initial search produced 2914 results. We divided these results such that the paper title and venue were read by at least two people, marking the paper as relevant, irrelevant or possibly relevant. Papers which the title readers agreed were not relevant were discarded, while the rest were moved on to the next stage, with a single reader reading the abstract for relevancy. In total 975 abstracts were read, with readers deciding that 317 of these papers were relevant or possibly relevant. A further round examined the paper introduction and optionally further paper details.

Papers included by both the snowballing process and systematic search were summarized in a shared table, recording source and target language, type of transformation, purpose of the paper, and research classification as per Wieringa et

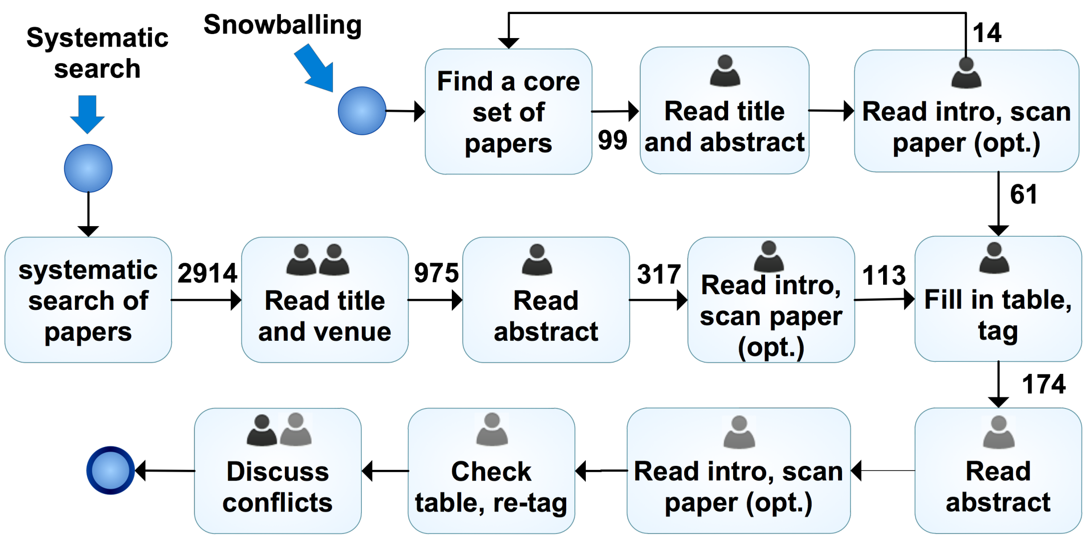


Fig. 1: A Summary of the Publication Finding, Exclusion, Summary and Tagging Process (counts indicate included papers after each step)

al. [33]. The summary included a set of tags (described in Sec. IV) summarizing the purpose and source/targets of the approach, derived via a Grounded Theory, grouping qualitative data according to relevant categories or codes relating to potentially interesting observations or theories [31]. For each type of paper, an additional reader was assigned to re-read the title, abstract, and summary of included papers, optionally looking at further paper details in order to re-tag the papers. Differences between tags were identified and discussed, resulting in a final set of tags. During this process, 23 duplicate publications, including overlaps between snowballing and systematic search papers, were identified. In total, our roadmap summarizes 174 papers.

IV. Results

In this section we present the results of our roadmap, using them to answer **RQ0-9**. We answer our overarching research question **RQ0** (What approaches exist which map/integrate/transform goal-oriented languages to/from other RE/SE software artifacts or languages?) by providing the full list of 174 publications, including paper name, authors, venue, publication year, and other summary data. We make this information available online [2]. We answer the more detailed **RQ1-9** in the following.

RQ1: Transformation Type. We report the number of papers which were classified under mapping, transformation or integration in Table IV, including a classification of horizontal vs. vertical. Counts for endogenous/exogenous vs. vertical/horizontal are in Table V. Note that it is possible for a paper to fall under more than one category, in which case it is counted for each category.

Techniques use both mappings and transformations, showing that both approaches are feasible even with highly social models such as goal models. We can note that there are more vertical transformations and more horizontal mappings. This may indicate that it is more feasible to develop transformations when moving downstream, decomposing models into further detail, than when moving horizontally, linking models to a view at the same level of abstraction. We can also see that most approaches focus on transformations or mappings, avoiding language integrations, likely in order to avoid creation of overly complex languages.

Examining Fig. V, transformations (including mappings/ integrations) are mostly vertical exogenous, with an apparent focus on moving goal models downstream via transformations to other artifacts. Despite the predominance of vertical exogenous transformations, we see a significant number of horizontal exogenous approaches, transforming goal models to other models at the same level of abstraction. Further exploration of horizontal transformations is provided as part of **RQ3**. Noting the predominance of exogenous transformations, we may surmise that although the intentional view provided by goal models is useful, most techniques see value in multiple conceptual views of the system, captured by multiple types of models. Issues encountered when classifying techniques as horizontal or vertical are discussed further in Sec. V.

RQ2: Goal model Source/Targets. In Table VI we list the top eight types of goal models used as targets, including the counts of the number of publications using each language. Note that we found that many papers used "goal models" in general, without referring to a specific existing language. We could interpret these counts as indicating whether or not a particular goal model framework is more or less amenable to transformations. Alternatively, these counts could attest to the popularity or level of adoption of various goal modeling frameworks.

[2]http://goo.gl/Zw6B3T

TABLE IV: TECHNIQUE COUNT CLASSIFIED AS MAPPING, TRANSFORMATION, OR INTEGRATION VS. HORIZONTAL OR VERTICAL

	Mapping	Transformation	Integration
Vertical	49	64	12
Horizontal	32	23	12

TABLE V: TECHNIQUE COUNT CLASSIFIED AS EXOGENOUS OR ENDOGENOUS VS. HORIZONTAL OR VERTICAL

	Endogenous	Exogenous
Vertical	9	110
Horizontal	16	47

TABLE VI: THE TOP 8 GOAL MODEL SOURCE LANGUAGES

Unidentified Goal Model	*i**	KAOS	Tropos	NFR	GRL	AOV	Map
57	46	23	17	8	5	4	4

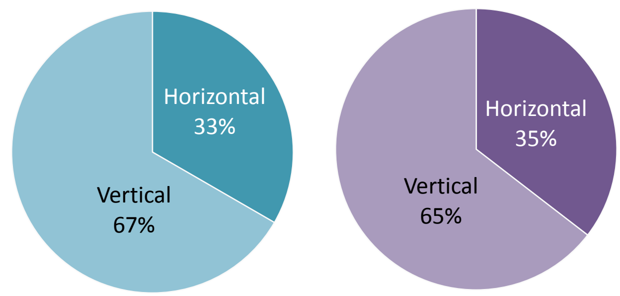
RQ3: Non-Goal Model Source/Targets. As described in Sec. III, we undertook a grounded theory process in order to classify and tag publications. Our purpose was to summarize the purpose and motivation and the means associated with the included publications. This process resulted in the creation of a taxonomy of tags, under the general category of "how", i.e. the source/target of the transformation and "why", the general purpose or paradigm of the approach.

Our results include 260 unique non-goal model source and targets (one publication can be classified as having multiple sources and targets, thus a count greater than 174). Sources and targets are listed as part of the "how" tag taxonomies in Figs. 2 and 3. The "how" taxonomy is divided into two views, focusing on approaches which integrate with or transform to goal models (Fig. 2) and which transform from goal models to some other target (Fig. 3). To help summarize our findings, we cluster detailed tags into more abstract categories (dashed boxes), e.g., Software Artifact, Requirements, and Business Artifact. As such classifications are subjective, our purpose is not to propose a rigorous hierarchy for system development, but instead to provide a higher-level, grounded summary of our results.

Counts are included in the upper right or in parentheses. Tags without a count have a default count of (1). Taggers could chose to classify leaf-level tags into more than one category, e.g., Activity Diagrams are UML Models and Behavioral Models. Words or phrases considered as pseudonyms for our tags are shown in parentheses, e.g., Business Rule is included under Informal Constraint.

Our taxonomies show that goal models have been transformed to and from a wide variety of languages and artifacts. Much activity has focused on transforming goal models to other modeling languages, particularly outside of UML (47 tag counts, listed in the middle left of in Fig. 3). 19 tags capture transformations from GOL to some form of UML. Several other approaches have focused on formalizing goal models (22 tags), producing system constraints (26 tags). We can see that Business processes models and architecture are other popular targets, with 19 and 27 tags, respectively.

Most activity (87%) transforms goal models to another artifact, consistent with the view of goal models as an artifact for early requirements. However, it is interesting to note, as shown in Fig. 2, that several techniques use goal models as a target language. We may guess that most of these techniques



(a) to Goal Models (b) from Goal Models

Fig. 4: Horizontal vs. Vertical Transformations in Techniques Transforming to/from Goal Models

may involve horizontal transformations, better exploring the requirements space. However, we can examine horizontal vs. vertical transformations in techniques which transform to and from goal models (Fig. 4, left and right, respectively). We can see that techniques transforming to goal models use vertical transformation nearly as often as those transforming from goal models. This indicates that techniques may be using goal models for purposes beyond early requirements, as artifacts downstream from other artifacts.

We can examine trends in the non-goal model source/targets of the approaches over time. Fig. 5 shows the frequency of the source/target mid-level categories over the last ten years (integrations with goal models and transformation to goal models in the top two charts, transformations from goal models in the bottom two charts). Note that the taxonomies in Figs. 2 and 3 cover several papers published before 2003 arising from the snowballing process, while Fig. 5 covers only 2003-2013. As such, the counts between the figures do not match exactly.

Looking at Fig. 5, we can observe some trends. In the top left chart, it seems that publications integrating another artifact or language with goal models has peaked, and is beginning to decline. The same could be said for techniques transforming goal models to requirements. There seems to be an overall increase in techniques transforming business artifacts and non-UML models to goal models; however, as the counts for transformations to goal models are very low overall, trends may not be significant.

Examining the bottom row of charts in Fig. 5, one can notice an increase in transformations from goal models to

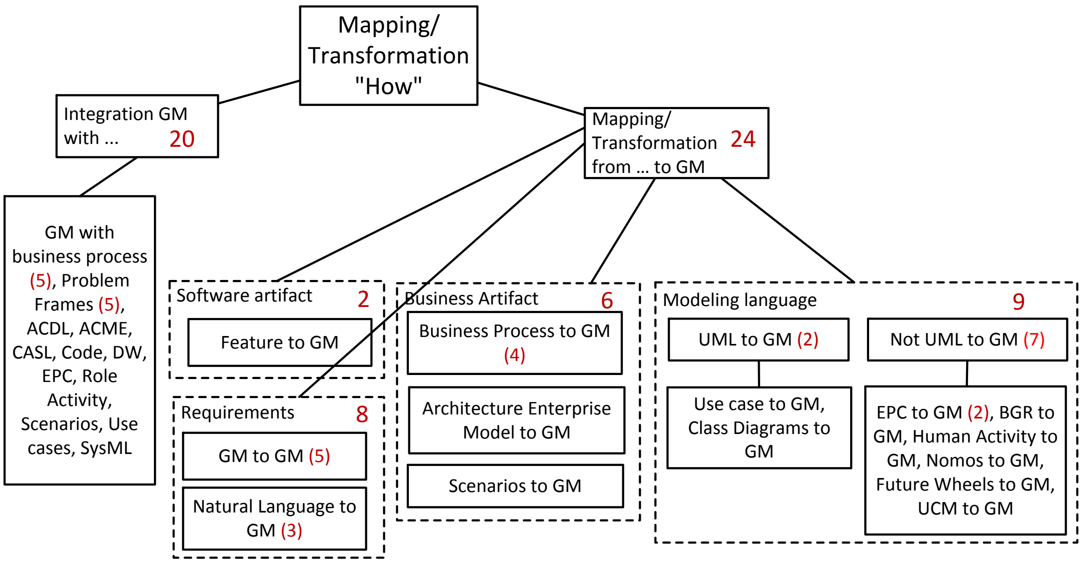


Fig. 2: Taxonomy of "how" Tags used to Classify Publications Transforming to Goal Models (default count of (1))

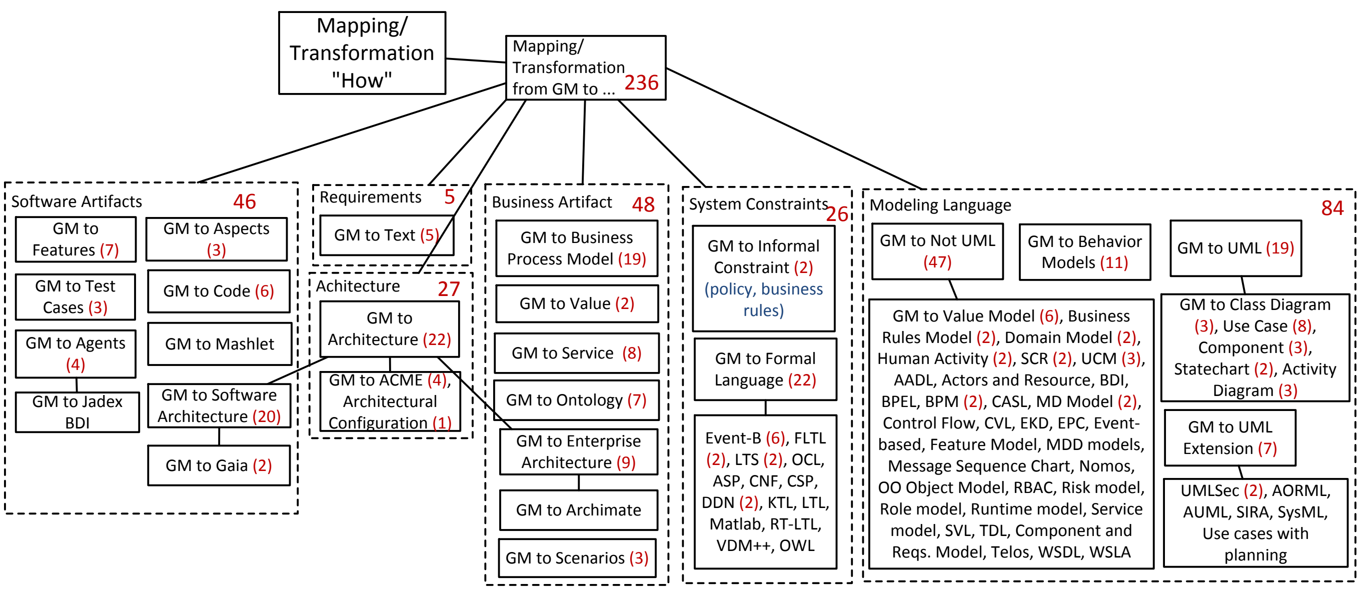


Fig. 3: Taxonomy of "how" Tags used to Classify Publications Transforming from Goal Models (synonyms in parentheses, default count of (1))

business artifacts, architecture, and to non-UML models. Effort in transformations to UML models seems to be in decline. Transformations to system constraints (formal or otherwise) appears to have peaked and is now also in decline. Transformations to software artifacts appears to be holding a steady pattern of peaking every few years.

RQ4: Technique Motivations. We show our "why" taxonomy in Fig. 6. Several of the publications used such paradigms in order to motivate work, e.g. "taking advantage of the benefits of aspect-orientation", with the assumption that the benefits associated with a particular paradigm were well-known. As such, tags in this taxonomy are not clearly motivations, but often describe general paradigms, e.g. service-orientation, aspect-orientation. As it is not our purpose to describe the potential benefits of such paradigms, we stop our "why" analysis at this level. As with the "how" taxonomy, we classify the "why" taxonomy into higher-level categories (dashed boxes), e.g., Enhanced RE, Business Analysis, and Decision Making.

We examine the frequency of leaf-level "why" tags over time in Fig. 7. Here we only show the most frequent 12 tags. We can note that use of goal model transformations

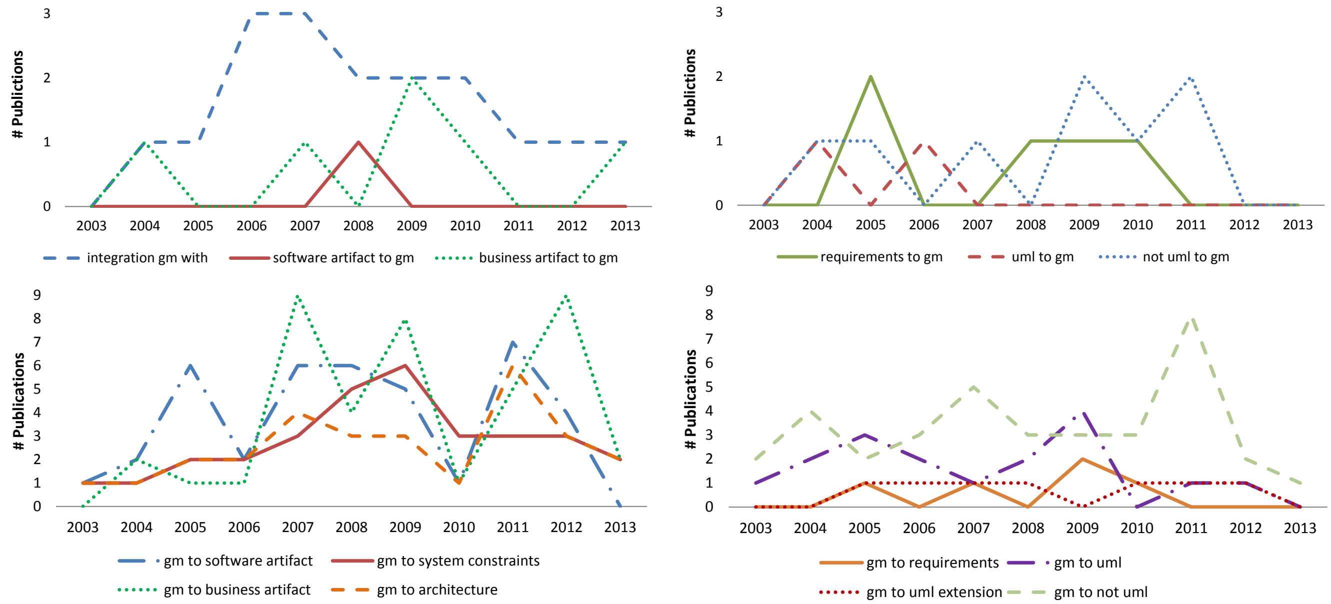


Fig. 5: Frequency of "How", Non Goal-Model Source/Target Categories over Time (Top Row: Integrations with Goal Models and Transformations from Goal Models, Bottom Row: Transformations to Goal Models)

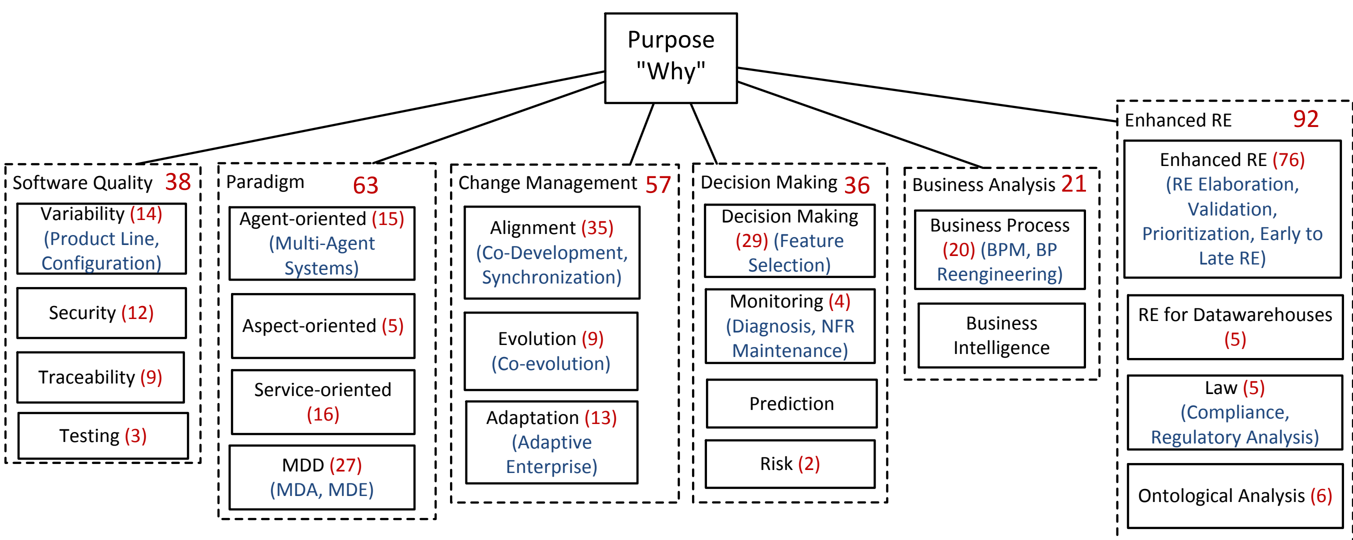


Fig. 6: Taxonomy of "why" Tags used to Classify Publications (synonyms in parentheses, default count of (1))

for Enhanced RE seems to be decreasing, while some other categories, e.g., Alignment, appear to be rising. The remainder of the categories do not show obvious trends.

RQ5: Research Classification. We have classified the publications under the research classifications described by Wieringa et al. [33], as follows: validation 3 papers (2%), evaluation 1 (1%), solution 157 (91%), philosophical 2 (1%), opinion 2 (1%), and experience 8 (5%). Publications could be classified into more than one category. In addition we found six papers which we classified as related surveys, described in Sec. VI. The high presence of solution papers was due in part to our practice of classifying papers as evaluation or experience only if this was the primary purpose of the paper, i.e. papers which presented a new method with some evaluation were classified only as solution. Even so, the prominence of solution papers can be interpreted as an indication of the immaturity of the field, with many proposals lacking extensive application or evaluation.

RQ6: Venue. We list the top 15 venues in Fig. 8, with 43% of included publications appearing in these 15 venues. Our results show that relevant publications appear in a total of 104 unique conference/book/journal venues. This wide spread

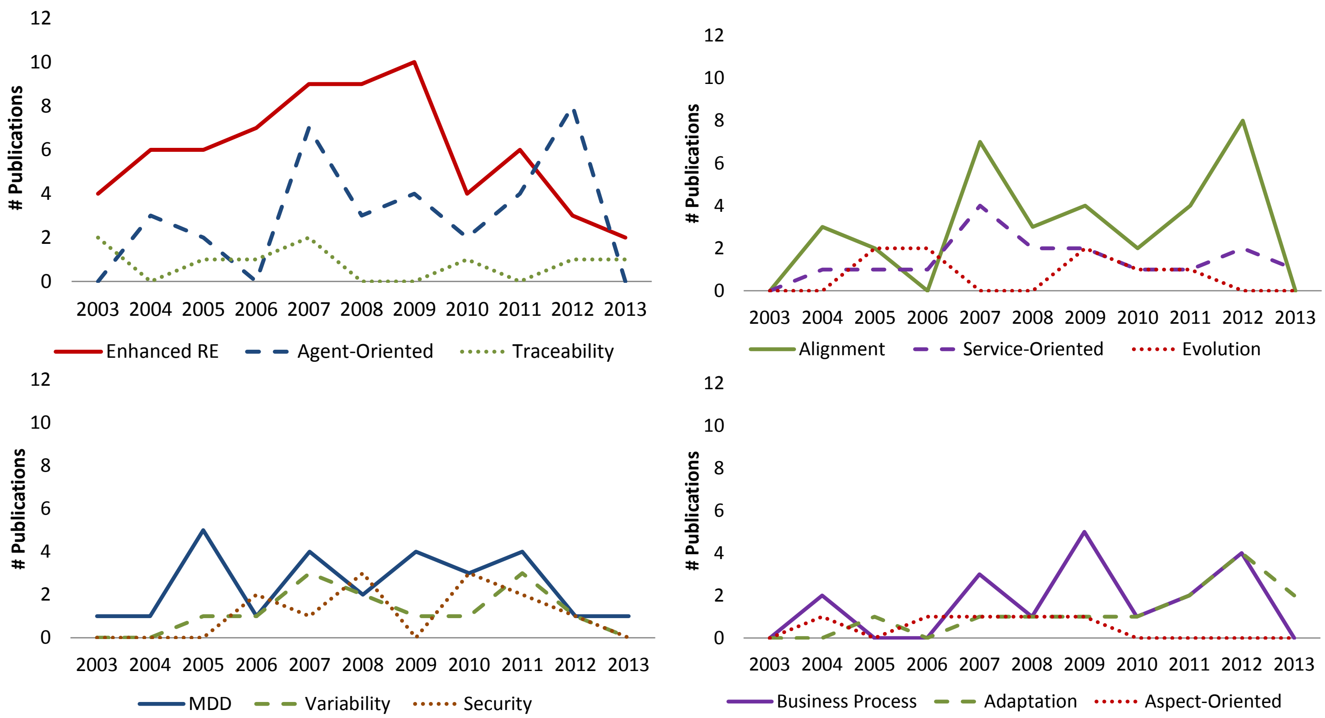


Fig. 7: Frequency of Most Common “Why” Leaf Tags over Time

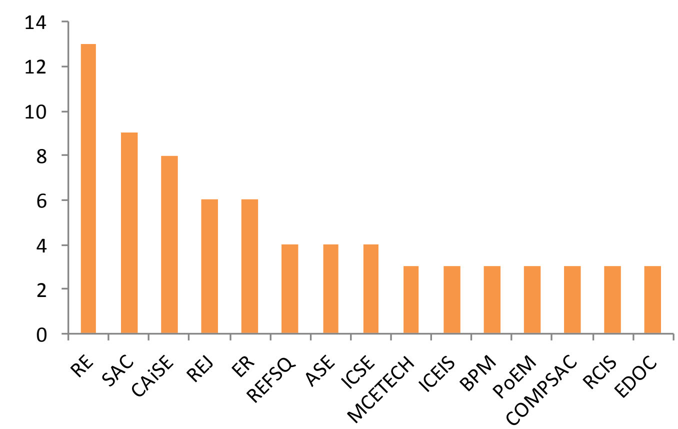


Fig. 8: Top 15 Publication Venues

in venues may hinder consolidation of research results, while helping dissemination to a wider audience.

RQ7: Citations. We show citations numbers for the top 30 cited papers in Fig. 9, listing the top five cited papers in the figure. Other papers can be identified via their identifier by looking at our online list of papers[3]. We see that all of the top five cited papers have been published at least nine years ago. Time, not surprisingly, is a significant factor in accumulating citations. Citations numbers are heavily weighted towards a few papers, although the average number of citations is 43, the top 10 cited papers have 63.5% of the total citations. We can conclude that although many approaches are proposed, most have not been extensively re-applied in a research context.

RQ8: Authors. Examining the 174 included publications, we see a total of 351 authors. Among the 351 authors, 104 have at least two publications included in our roadmap, while 51 have at least three publications. We can use data on the included papers to create views of the co-author networks of paper authors. We show a high-level complete view of the authors of all 174 papers in Fig. 10, a more readable version of this model can be found online[4]. One can note a few large, strongly-connected clusters of co-authors and the presence of many small clusters indicating that the level of collaboration is still relatively low. A more detailed view of authors who have more than three included publications is shown in Fig. 11.

RQ9: Interest. We have asked “Is interest in goal model integration increasing or decreasing?” Fig. 12 shows the number of publications per year in our mapping, including an interpolation line. We can see a rise from 2003 to 2007, with peaks in 2007, 09, and 11. The tendency towards publication in this area seems to be decreasing, although results for 2013 are only complete up to early fall.

V. Threats to Validity

We can identify several threats to the validity of our study.

[3]http://goo.gl/Zw6B3T

[4]http://goo.gl/hDCcQa

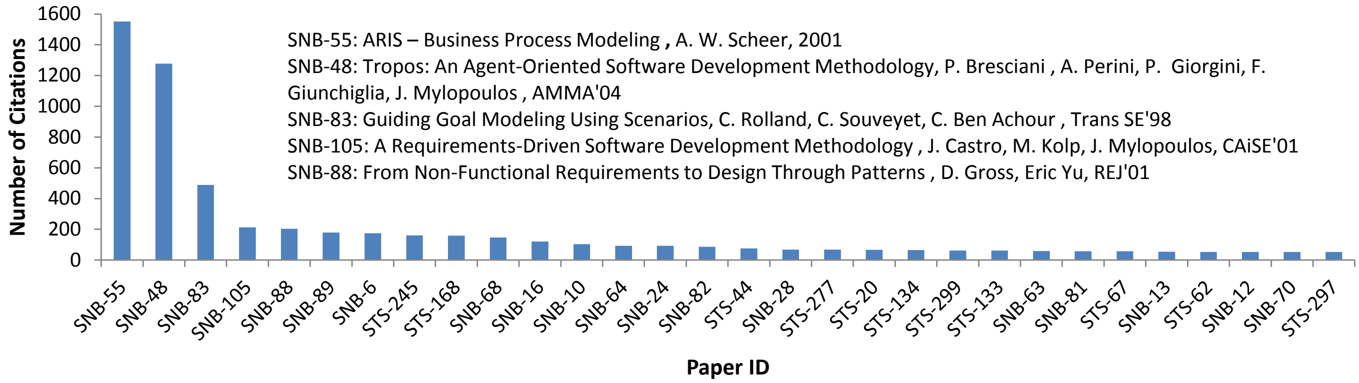


Fig. 9: Citations Counts for the Top 50 Publications

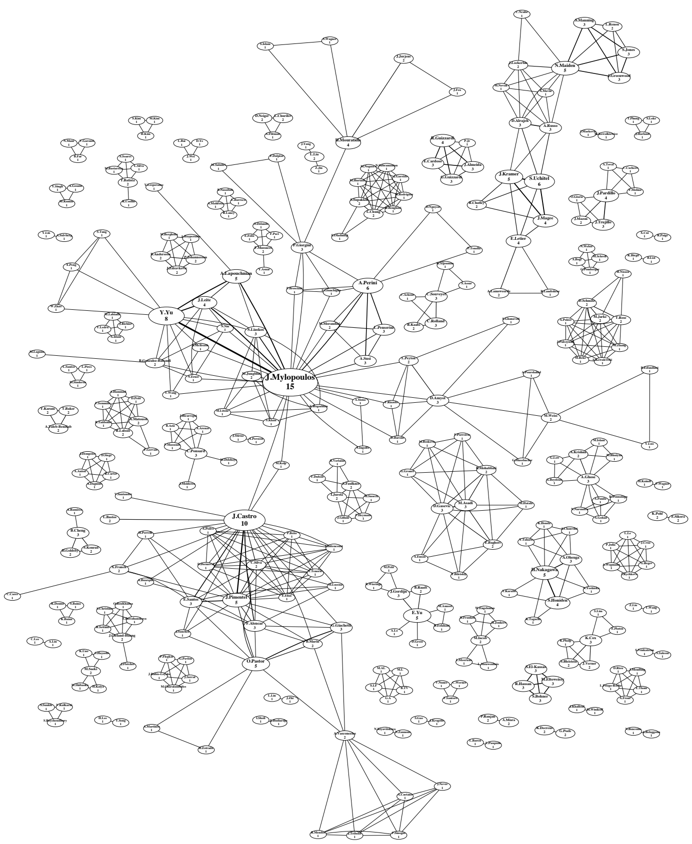


Fig. 10: Co-author Network - Complete High-Level View

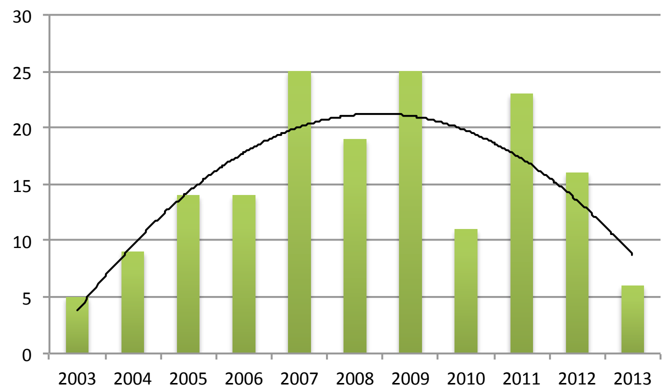


Fig. 12: Included Papers per Year

Study Completeness. Although we have covered 174 papers through systematic search and snowballing, our study results are likely not complete, threatening *Conclusion Validity*. As of this reporting of our results, we have not yet finished the process of snowballing, finding references through known related work. We have further opted not to snowball over publications found through systematic search. Our experience echos an issue highlighted by Kitchenam et al. [21], the high amount of effort required to undertake a Systematic Literature Review (SLR). We plan to increase the completeness of our roadmap in follow-up studies.

Our systematic search criteria may also be subject to critique, threatening *Construct Validity*. Initially, we hoped to include goal synonyms in our search string (e.g., intention, motivation); however, as many papers use these words outside the context of goal modeling, our search returned more than 10,000 results.

We had expected to find a higher number of overlapping papers between our systematic and snowballing results. This may indicate that our search string was not an effective representation of our desired scope, or it may be due to our incomplete snowballing process.

Several threats relate to the process of including, summarizing, and tagging papers, i.e. the *Internal Validity* of our results.

Publication Inclusion. The inclusion or exclusion of papers in our survey may be subjective or error prone. We have tried to mitigate this threat by defining and using clear inclusion and exclusion criteria (see Table II), and by having at least two people read paper titles for relevance, with disagreements checked by reading the abstract and (optional) paper details. We undertook group discussions to decide on the inclusion or exclusion of several papers whose status was uncertain.

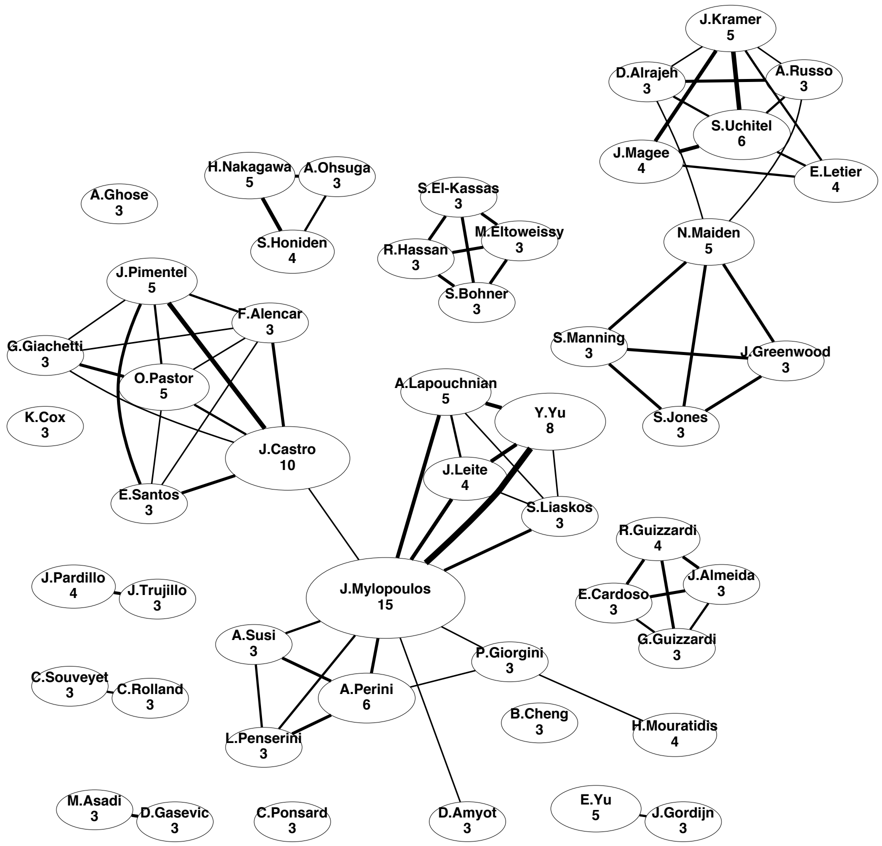


Fig. 11: Co-author Network - Detailed View of Co-authors with more than Three Included Papers

However, our process may lean towards exclusion, as included papers were checked several times for inclusion, but papers excluded beyond the title stage were not always checked by another reader.

Publications Summary. The summary information collected for papers could also be subjective. For example, the classification of approaches into horizontal or vertical transformations was particularly difficult, as well the distinction between extension and integration. We addressed these issues through group discussion, identifying clear examples of horizontal vs. vertical transformations and language extension vs. integration.

Tagging. Our tagging classifications may also be subjective, especially when moving away from the leaf tags in the taxonomy. We mitigated this issue in part by re-tagging publications using an agreed-upon taxonomy with clear paths up to more general tags. We have opted not to collect a formal method of reader or tagger agreement (e.g., Kappa measure). Instead of aiming for a high-level of initial agreement between readers and taggers, our strategy was to resolve ambiguous cases via group discussion. Although we noted our initial agreement for tags, especially "why" tags was quite high (we disagreed on more than half of the papers) all tags eventually converged via pair-wise author discussion.

Author Experience. The paper authors have significant experience in goal modeling (typically *i**-related languages). This helps to increase our confidence in the size of the initial core set of papers, but may also bias the survey coverage, i.e. threatening *External Validity*. As shown by the author network analysis, several authors of this paper are authors of papers covered via our roadmap. The chances of us considering our own papers for inclusion, especially through the snowballing process, are high. However, Figs. 2 and 3 show that the surveyed publications cover a wide variety of source/target goal modeling languages, while Figs. 10 and 11 show the inclusion of many authors beyond the authors of this paper. Such coverage is reflected both in our snowballing and systematic search results.

VI. Related Work

Literature Reviews in SE. We have created our roadmap by adopting the methods and approaches prescribed by Petersen et al. [29], specifically focusing initially on a roadmap of available work, rather than a detailed survey, clearly defining our process of finding and including papers, making our research questions clear. Kitchenham et al. provide guidelines for empirical studies in software engineering, we apply many of these guidelines where applicable to our systematic mapping study, including clearly specifying a hypothesis (in our case research questions), defining populations (publications from snowballing or systematic search of specific databases), defining a process, providing raw data, and making extensive use of graphics [20].

Work by Kitchenam et al. performs a type of systematic "meta" review by reviewing and mapping SLRs in software engineering [23]. This work argues that existing studies often fail to assess the quality of the surveyed publications. In our case, as we focus only on mapping publications, we have not undergone any explicit assessment of paper quality. We leave this task to future work as part of a planned systematic review of the papers in our roadmap. Further work by some of the same authors evaluates the utility of roadmap (mapping) studies specifically [24]. Here, the authors emphasize making the mapping results available, specifically the classification of each paper, allowing for study follow-up by other authors. In our case, we make such data available online.

Work by Pham et al. focuses on a social network analysis of computer science publications, investigating collaboration and citations [30], applying such analysis to the CAiSE conference series in [19]. Pham et al. rank authors by using the Page Rank algorithm, which mainly considers how the authors' work is cited by others, while the author analysis included in Sec. IV ranks authors according to the number of their included publications. Pham et al. also color nodes according to (sub)community, while this information is not yet an output of our roadmap analysis.

Related Literature Reviews. Our roadmapping process found other SLRs which cover topics related to our scope and research questions. As these approaches are literature reviews and not roadmaps, they have a deeper analysis of relevant papers, but have a much narrower focus on fewer papers. Decreus et al. look at six techniques transforming *i** to business process models [12], while Assar & Souveyet review the use of the goal concept in eight approaches for web service discovery [5]. Galaster et al. evaluate current approaches which address the gap between requirements and architecture, including several approaches starting from goals-oriented languages [14]. In [15] the same authors create a method for assessing and comparing approaches for transitioning from requirements to architecture. They use their method to assess 14 methods, two of which start from goal models. Work in [6] considers the suitability of five existing goal-oriented frameworks for modeling strategic alignment from a management information systems perspective, using concepts from strategy maps to evaluate the suitability of goal models for this purpose.

In [3], Amyot & Mussbacher perform a SLR of publications, finding 281 using the User Requirements Notation (containing the Goal-oriented Requirement Language (GRL)). The paper classifies surveyed work into 17 categories, several of which (e.g., Web Applications and Web Services, Transformations to Design Models, Feature Interaction Analysis, Aspect-oriented Modeling) fall under the scope of our survey. Our approach differs from this work by focusing more broadly on all goal-oriented languages and more narrowly on transformations, while presenting results as part of a roadmap, instead of a detailed comparison and evaluation.

VII. Discussion and Conclusions

We have conducted a systematic study creating a roadmap of publications which transform goal modeling language to or from other software artifacts and/or models, summarizing publication details and trends answering **RQ0-9**. More specifically, the final 174 publications included in our roadmap provide a variety of transformations involving goal models, especially transformations from goal models to other models/artifacts, with a high number of vertical, exogenous approaches. Only a small percentage of publications are widely cited, and most work focuses on new solutions, instead of solution evaluation or application experience. We observe that many approaches are narrowly focused, with most approaches focusing only on a few stages of the software lifecycle, not often providing an end-to-end solution. Analysis of the network of coauthors shows that authorship in this area is still relatively fragmented, with many small isolated groups. Although the number of publications in this area has increased in the last 10 years, the interest seems to have peaked. However, the frequency of solution papers leads us to believe that work in this area is still relatively immature, with a divergent set of approaches.

Although the there are many examples of successful industrial applications of goal models (e.g., [26], [1]), most efforts have a high degree of academic participation - practitioners do not often adopt goal-oriented techniques on their own initiative. The lack of widespread industrial adoption could be attributed to several factors, for example, the presence of many competing goal-oriented languages and tools lacking standardization, or scalability and usability challenges in complex models which are not easily decomposable.

One may hypothesize that a further barrier may be difficulties in integrating goal models with other system artifacts. However, the results of our roadmap have shown that many techniques have been proposed to facilitate this integration. Despite the availability of techniques, the divergence and lack of evaluation for methods in this area are likely to discourage independent adoption.

Although progress has been made on transforming social-oriented models to other artifacts, a deeper analysis of available work is needed to understand what concepts within goal models are frequently involved in the transformations, and whether the more qualitative concepts such as softgoals and roles are often mapped or ignored.

We have performed some initial social network analysis on authorship in this area. However, a deeper analysis examining the reference structure between papers, determining what techniques build on other techniques, may be of greater interest.

Future work over the data provided in this paper should continue in two directions, to expand the roadmap to cover more publications, particularly through snowballing, and to look more closely at a selected subset of papers, performing a more detailed review evaluating paper content and quality.

Acknowledgments

This work was supported in part by ERC advanced grant 267856, titled “Lucretius: Foundations for Software Evolution”, http://www.lucretius.eu.

References

[1] (2011) istar showcase’11: Exploring the goals of your systems and businesses, practical experiences with i* modeling. [Online]. Available: http://www.cs.toronto.edu/km/istar/iStarShowcase_Proceedings.pdf

[2] F. Alencar, B. Marin, G. Giachetti, O. Pastor, J. Castro, and J. H. Pimentel, “From i* requirements models to conceptual models of a model driven development process,” in *The Practice of Enterprise Modeling*, 2009, vol. 39, pp. 99–114. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-05352-8_9

[3] D. Amyot and G. Mussbacher, “User requirements notation: the first ten years, the next ten years,” *Journal of Software*, vol. 6, no. 5, pp. 747–768, 2011. [Online]. Available: http://ojs.academypublisher.com/index.php/jsw/article/view/0605747768

[4] A. Anton and C. Potts, “The use of goals to surface requirements for evolving systems,” in *Proc. of the 20th International Conference on Software Engineering*, 1998, pp. 157–166.

[5] S. Assar, C. Souveyet, and K. Aljoumaa, “A goal-oriented perspective on approaches to web service discovery,” in *Proc. of the 36th IEEE Annual Computer Software and Applications Conference Workshops (COMPSACW12)*, 2012, pp. 347–352.

[6] A. Babar, B. Wong, and A. Gill, “An evaluation of the goal-oriented approaches for modelling strategic alignment concept,” in *Proc. of the 5th International Conference on Research Challenges in Information Science (RCIS11)*, 2011, pp. 1–8.

[7] L. Bastos and J. Castro, “From requirements to multi-agent architecture using organisational concepts,” *SIGSOFT Software Engineering Notes*, vol. 30, no. 4, pp. 1–7, 2005. [Online]. Available: http://doi.acm.org/10.1145/1082983.1082980

[8] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos, “Tropos: An agent-oriented software development methodology,” *Autonomous Agents and MultiAgent Systems*, vol. 8, no. 3, pp. 203–236, 2004.

[9] L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos, *Non-functional requirements in software engineering*, ser. International Series in Software Engineering. Kluwer Academic Publishers, 2000, vol. 5.

[10] K. Czarnecki and S. Helsen, “Classification of model transformation approaches,” in *Proc. of the 2nd OOPSLA’03 Workshop on Generative Techniques in the Context of Model-Driven Architecture*, 2003.

[11] A. Dardenne, A. Lamsweerde, and S. Fickas, “Goal-directed requirements acquisition,” *Science of Computer Programming*, vol. 20, no. 1-2, pp. 3–50, 1993.

[12] K. Decreus, M. Snoeck, and G. Poels, “Practical challenges for methods transforming i* goal models into business process models,” in *Proc. of the 17th IEEE International Requirements Engineering Conference (RE09)*, 2009, pp. 15–23.

[13] H. Estrada, A. Martínez, and O. Pastor, “Goal-based business modeling oriented towards late requirements generation,” in *Proc. of the 22nd International Conference on Conceptual Modeling (ER03)*, 2003, vol. 2813, pp. 277–290. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-39648-2_23

[14] M. Galster, A. Eberlein, and M. Moussavi, “Transition from requirements to architecture: A review and future perspective,” in *Proc. of the 7th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD06)*, 2006, pp. 9–16.

[15] ——, “Comparing methodologies for the transition between software requirements and architectures,” in *IEEE International Conference on Systems, Man and Cybernetics (SMC09)*, 2009, pp. 2380–2385.

[16] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone, “Modeling security requirements through ownership, permission and delegation,” in *Proc. of the 13th IEEE International Conference on Requirements Engineering, 2005*, 2005, pp. 167–176.

[17] D. Harel and B. Rumpe, “Meaningful modeling: What’s the semantics of "semantics"?” *Computer*, vol. 37, no. 10, pp. 64–72, Oct. 2004. [Online]. Available: http://dx.doi.org/10.1109/MC.2004.172

[18] ITU-T and Intern. Telecom. Union: Recommendation Z.151 (11/08), “Recommendation Z.151 (11/08) User Requirements Notation (URN) – Language definition,” http://www.itu.int/rec/T-REC-Z.151/en., ITU-T – International Telecommunications Union, Tech. Rep., 2008.

[19] M. Jarke, M. Pham, and R. Klamma, “Evolution of the caise author community: A social network analysis,” in *Seminal Contributions to Information Systems Engineering*, 2013, pp. 15–33. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-36926-1_2

[20] B. Kitchenham, S. Pfleeger, L. Pickard, P. Jones, D. Hoaglin, K. Emam, and J. Rosenberg, “Preliminary guidelines for empirical research in software engineering,” *IEEE Transactions on Software Engineering*, vol. 28, no. 8, pp. 721–734, 2002.

[21] B. Kitchenham and P. Brereton, “A systematic review of systematic review process research in software engineering,” *Information and Software Technology*, vol. 55, no. 12, pp. 2049 – 2075, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950584913001560

[22] B. Kitchenham, T. Dyba, and M. Jorgensen, “Evidence-based software engineering,” in *Proc. of the 26th International Conference on Software Engineering (ICSE04)*, 2004, pp. 273–281. [Online]. Available: http://dl.acm.org/citation.cfm?id=998675.999432

[23] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, “Systematic literature reviews in software engineering – a tertiary study,” *Information and Software Technology*, vol. 52, no. 8, pp. 792 – 805, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950584910000467

[24] B. A. Kitchenham, D. Budgen, and O. P. Brereton, “Using mapping studies as the basis for further research – a participant-observer case study,” *Information and Software Technology*, vol. 53, no. 6, pp. 638 – 651, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950584910002272

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

[26] A. V. Lamsweerde, “Goal-oriented requirements engineering: a roundtrip from research to practice,” in *Proc. of the 12th IEEE International Requirements Engineering Conference*, 2004. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.67.6087\&rep=rep1\&type=pdf

[27] L. Liu and Z. Jin, “Integrating goals and problem frames in requirements analysis,” in *Proc. of the 14th IEEE International Conference Requirements Engineering*, 2006, pp. 349–350.

[28] T. Mens and P. Van Gorp, “A taxonomy of model transformation,” *Electronic Notes in Theoretical Computer Science*, vol. 152, pp. 125–142, 2006. [Online]. Available: http://dx.doi.org/10.1016/j.entcs.2005.10.021

[29] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, “Systematic mapping studies in software engineering,” in *Proc. of the 12th international conference on Evaluation and Assessment in Software Engineering (EASE08)*, 2008, pp. 68–77. [Online]. Available: http://dl.acm.org/citation.cfm?id=2227115.2227123

[30] M. Pham, R. Klamma, and M. Jarke, “Development of computer science disciplines: a social network analysis approach,” *Social Network Analysis and Mining*, vol. 1, no. 4, pp. 321–340, 2011. [Online]. Available: http://dx.doi.org/10.1007/s13278-011-0024-x

[31] C. Seaman, “Qualitative methods in empirical studies of software engineering,” *IEEE Transactions on Software Engineering*, vol. 25, no. 4, pp. 557–572, 1999.

[32] R. Sebastiani, P. Giorgini, and J. Mylopoulos, “Simple and minimum-cost satisfiability for goal models,” in *Advanced Information Systems Engineering*, ser. Lecture Notes in Computer Science. Springer, 2004, vol. 3084, pp. 20–35. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-25975-6_4

[33] R. Wieringa, N. Maiden, N. Mead, and C. Rolland, “Requirements engineering paper classification and evaluation criteria: a proposal and a discussion,” *Requirements Engineering*, vol. 11, no. 1, pp. 102–107, 2006. [Online]. Available: http://dx.doi.org/10.1007/s00766-005-0021-6

[34] E. Yu, “Towards modelling and reasoning support for early-phase requirements engineering,” in *Proc. of the 3rd IEEE International Symposium on Requirements Engineering*, vol. 97, 1997, pp. 226–235.