Can Patterns improve i* Modeling? Two Exploratory Studies

Markus Strohmaier[1], Jennifer Horkoff[2], Eric Yu[3], Jorge Aranda[2],and Steve Easterbrook[2]

[1] Knowledge Management Institute, Graz University of Technology, Inffeldgasse 21a, Graz, Austria
[2] Department of Computer Science, University of Toronto, 10 King’s College Road, Toronto, Ontario, Canada
[3] Faculty of Information Studies, University of Toronto, 140 St. George St., Toronto, Ontario, Canada
markus.strohmaier@tugraz.at, yu@fis.utoronto.ca, {jenhork, jaranda, sme} @cs.utoronto.ca

Abstract. A considerable amount of effort has been placed into the investigation of i* modeling as a tool for early stage requirements engineering. However, widespread adoption of i* models in the requirements process has been hindered by issues such as the effort required to create the models, coverage of the problem context, and model complexity. In this work, we explore the feasibility of pattern application to address these issues. To this end, we perform both an exploratory case study and initial experiment to investigate whether the application of patterns improves aspects of i* modeling. Furthermore, we develop a methodology which guides the adoption of patterns for i* modeling. Our findings suggest that applying model patterns can increase model coverage, but increases complexity, and may increase modeling effort depending on the experience of the modeler. Our conclusions indicate situations where pattern application to i* models may be beneficial.

Keywords: The i* Framework, Model Patterns, Modeling Effort, Model Coverage, Model Complexity

1 Introduction

In the field of requirements engineering, much work has been dedicated to modeling in the early stages of the requirements engineering process. Models created in the i* Framework capture the goals of stakeholders and help requirements engineers to understand the strategic interactions and dependencies among agents [20]. These models are assumed to, for example, facilitate analysis and discover new knowledge about the domain. However, widespread adoption of such models in the requirements engineering process has been hindered by a series of issues [7], including:

Costs of modeling: The effort necessary to create, maintain, understand, and analyze i* models is high.

Model coverage: Due to the high complexity of social relations, i* models may fail to cover all relevant issues.
Complexity of models: At the same time, the models that result from modeling with the i* Framework can be complex and difficult to scale.

Improving some of these aspects would represent an improvement to i* modeling practices. Usage of patterns in previous work suggests that patterns in general can provide, among others, the following benefits ([1],[3],[4]):

Reuse: By abstracting and packaging domain knowledge in a structured way, patterns enable the reuse of knowledge.
Modularization: Because patterns have a clearly defined focus and well defined areas of application, they contribute to modularizing the domain.
Communication: By providing an agreed upon vocabulary of domain knowledge, patterns facilitate communication among stakeholders.

Although there have been some initial efforts in using patterns for agent-oriented, social focused modeling ([15], [17]), patterns have not yet been applied extensively in this area. This might be because the use of patterns in this area brings new challenges related to pattern construction, selection, adaptation and evaluation, whose effects might cancel out the benefits of patterns application. Our research sets out to explore challenges related to patterns in an early requirements context.
Of the many different types of patterns which can be constructed, we are especially interested in the utility of model patterns, that is, patterns that capture knowledge for reuse in the form of conceptual models rather than textual descriptions. Specifically, we define i* model patterns as i* models which are generalizations of a particular domain or situation of interest, which can then be contextualized when applied to a more specific situation. In this work we focus on those patterns which describe a specific software or technology, capturing its characteristics and the roles involved in using it in an abstract and reusable way. We focus on these types of patterns because they relate to the challenges and typical use of i*, specifically, enabling the evaluation of a particular technological solution in a specific context. To acquire a deeper understanding about the effects of pattern use in this context we have raised a set of research questions and conducted both an exploratory case study and an initial experiment involving the construction, application and evaluation of i* model patterns. A methodology for i* pattern application is introduced as part of an exploratory case study in Section 5.

2. Patterns in the i* Framework

Although the application of patterns shows promise in addressing several of the issues associated with i* modeling, i* model patterns as we have defined them differ from patterns typically seen in later stage requirements and software engineering. These differences often involve form, (textual versus graphical representation), and focus, with i* models focusing on high-level solutions in the early stages of requirements analysis. In the area of requirements engineering, patterns have been used to capture and organize knowledge about requirements and requirements engineering techniques. In software engineering, pattern theory defines a pattern as a construct that captures some proven knowledge of a domain via problem/context/solution triples which are created for further reuse [1]. In contrast, model patterns expressed in i* focus on the social context and interactions of the pattern subject matter. The patterns used in this work capture general requirements for the technology in the form of goals and softgoals as well as general goals of the roles it interacts with, including the dependencies between them.

We assume familiarity with the i* Framework [21]. As an example of an i* model pattern, consider a pattern describing the social relationships surrounding the usage of a wiki, as shown in Fig. 1. This model is a pattern as it describes a typical usage setting for a wiki. We expect that particular instances of wikis (expressed as *contextual models*) would have many of the features depicted in this model pattern, but possibly also deviations from it.

The wiki, as a technology system, is modeled as an agent. Its main task is to Provide for Mass Collaborative Authoring. It exists in the context of a number of roles – visitors, editors, reviewers, as well as a,Technology “Champion”, who wants to promote the benefits of the wiki. The champion depends on the wiki to achieve the goal Content be Correct/Useful as part of facilitating Collaborative Authoring. Each actor (agent or role) has its own goals and tasks and softgoals (success criteria), but ultimately depend on each other to form a social network. In this example, and in our pattern application methodology, the check marks in the model are used to indicate the extent to which the actor’s goals are achieved, using a procedure described in [13].

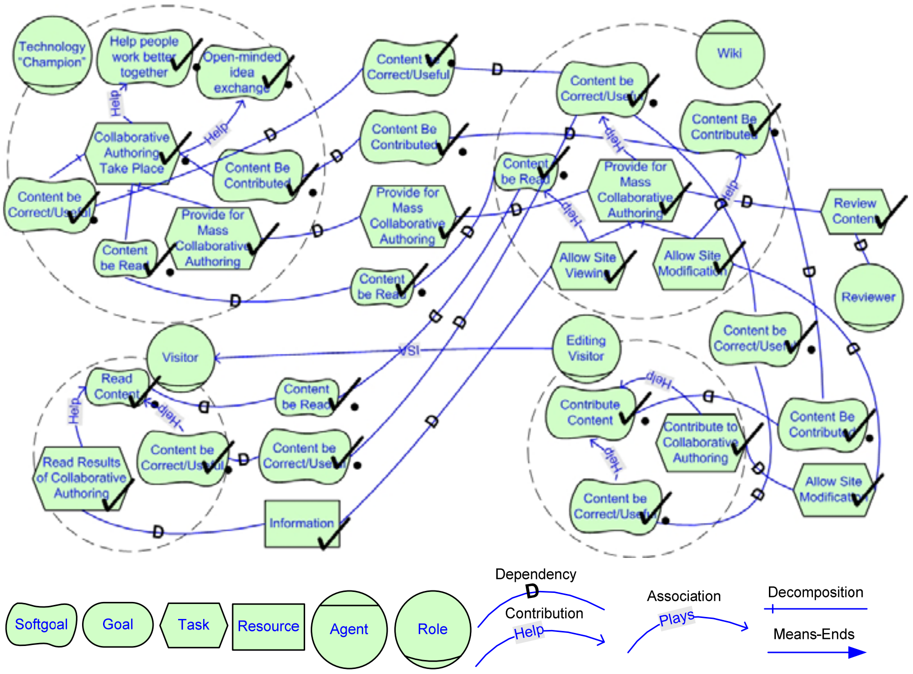


Fig. 1. Simplified Version of the Wiki Pattern

3. Research Questions and Research Design

Claims that the adoption of patterns improves requirements engineering efforts are abundant in literature, but the actual - positive or negative - effects that patterns have on requirements engineering in general and i* modeling in particular have not been studied in detail. Therefore, the overarching question of this paper is "*Can patterns improve modeling with the i* Framework?*" In order to clarify what we mean by "improve" and to make this question amenable to scientific investigations, we formulated a set of more specific questions. This work does not aim to find definite answer to these questions, but instead aims to explore their significance and sharpen their formulation.

Q1: Do model patterns help reduce modeling effort? Because i* model patterns are designed with reuse in mind, model patterns should contribute to decreasing the effort involved in i* modeling.
Q2: Do model patterns help increase model coverage? By capturing and documenting deep domain knowledge, the utilization and combination of i* model patterns should increase the degree to which i* models cover relevant aspects of the world.
Q3: Do model patterns help decrease complexity? Because i* model patterns have a scope and clearly defined borders, they should help to make the high complexity of i* models more manageable through modularization.

In order to investigate the positive and/or negative effects of patterns on i* modeling we need to observe instantiations of the modeling process. For this reason, we employ a research approach which uses a case study as well as an exploratory experiment to study the introduced research questions. The case study involves an ongoing requirements analysis project with an external organization, while the follow-up experiment, designed to address some of the limitations of the initial case study, uses student participants in a classroom setting.

5. Case Study: Kids Help Phone

In order to investigate our research questions concerning the use of patterns, we developed a methodology, or series of concrete steps, that guides and constrains the application of model patterns to i* models. In this section we outline the general steps of our proposed methodology, provide a description of the execution of these steps in the Kids Help Phone (KHP) case study, and present selected results.

5.1. Case Study Context and Preparation

This study uses data from an ongoing requirements analysis project with a not-for-profit youth counseling organization. KHP is a charitable, non-governmental organization that provides 24/7 counseling to kids across Canada via phone and web. The project was aimed to explore the situational "effectiveness" of a range of social technologies, such as discussion forums and wikis in their operations. To create an empirical baseline for our investigations, we interviewed stakeholders at KHP and constructed i* models of the domain without focusing on patterns. Specifically, we interviewed a total of ten stakeholders on their issues with knowledge transfer, in interview sessions lasting approximately one hour. The interviews acted as a basis for creating models that focused on the current usage of different technologies, such as a discussion forum. Finally, we assessed the current situation of KHP by evaluating the created models as a baseline for analyzing alternative solutions. In the case study, we chose to focus on a model representing the usage of a discussion forum.

5.2. Methodology and Case Study Execution

The left side of Fig. 2 provides a high-level overview of the steps involved in our methodology: from pattern creation, insertion and integration to the final evaluation of the resulting model. The right side of this figure contains some corresponding quantitative results of the study, explained in future sections.

5.2.1. Pattern Creation

This step involves the creation of i* model patterns. This will not be necessary once a patterns catalog becomes available.

Create Patterns. Create a set of patterns by consulting relevant literature. Model the roles, goals, tasks, resources, dependencies and contribution links related to a specific technology.

Case Study Application. In order to be able to evaluate a pattern approach in our case study, we created two model patterns – one pattern containing the use of wikis and one containing a discussion forum (Disc. F.). Application of the first pattern represents the case where the original technology in the domain (a discussion forum) is replaced by a new technology (a wiki), and application of the second pattern represents the case where a representation of an existing technology is replaced by a more detailed, generalized model of this technology.

Evaluate Patterns. Evaluate the model patterns, (using the qualitative procedure described in [13]), in order to ensure that the goals of the pattern are, in principle, achievable in certain scenarios.

Case Study Application. Both the wiki and discussion forum patterns were evaluated in light of various common implementation scenarios. See Fig. 1 for a simplified version of the wiki pattern containing an evaluation of stakeholder goals.

5.2.2. Pattern Application

1. Select Patterns. Select patterns which are believed to be applicable and beneficial in the contextual model. Compare the contributions of goals in the pattern to the goals expressed in the contextual model(s) for an indication of pattern applicability.

Case Study Application. In the case of KHP, we chose the two patterns we had previously created.

2. Contextualize the Pattern. View the selected pattern in light of the contextual model domain, adding and removing relevant and irrelevant links and elements.

Case Study Application. We contextualized the wiki model pattern, removing 7 of the 117 elements and 16 of the 169 links. In the case of the discussion forum model pattern, all elements and design options were considered relevant and no changes were made.

3. Insert the Pattern. Insert the pattern into the contextual model view.

Case Study Application. In each case, the model pattern was pasted into the contextual model file containing the discussion forum.

4. Linking Actors. Link the actors defined in the pattern to the actors in the contextual model.

Case Study Application. In the case of the wiki model pattern, we replaced the discussion forum of the contextual model with the wiki model pattern. The pattern contained roles such as Visitor, Editing Visitor or Technology "Champion", which were linked to the existing roles in the contextual model via the i* framework's actor association links (such as "PLAYS" and "IS-A").

5. Pattern Integration. Integrate the pattern into the contextual model.

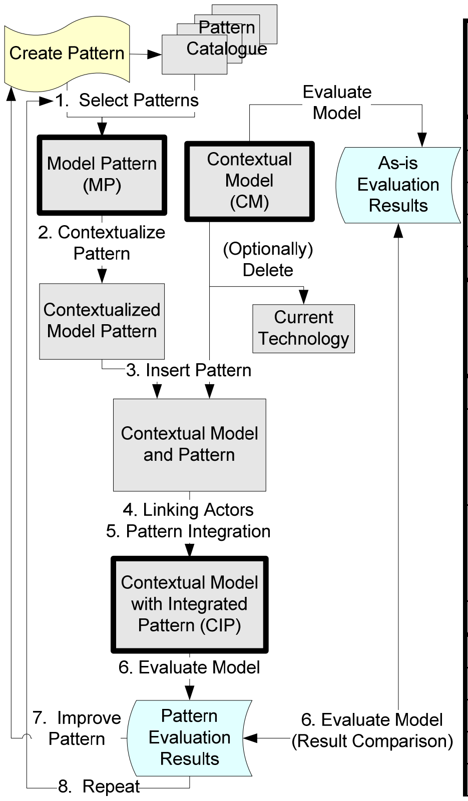
Case Study Application. The interactions between the pattern and the contextual actors were considered by adding or changing existing dependency links. The domain actors depended on their new roles in order to satisfy their goals, and, conversely, the technology agent depends on these actors, possibly indirectly, to be successful. In addition, we changed the existing elements and links in order to connect the new technology to the goals of existing actors.

We use measurements of model size and model changes as a way of quantifying our observations in relation to our research questions, see "Threats to Construct Validity" in Section 8 for a discussion of these measurements. As the first two points of measurement ("MP", "CM"), we considered the size of the contextual model and patterns before pattern insertion and integration. As the third point of measurement ("CIP") we considered the size of the model after all pattern integration changes were made. During the integration process, the number of i* constructs added, deleted or changed in some way was recorded. A summary of the measurements appears in the table on the right side of Fig. 2. Note that the differences between the top and bottom size counts in the table do not balance with the changes reported in the middle, as the measurements for steps 2 and 3 are not reported.

6. Evaluate Model. Evaluate and analyze the resulting model to determine whether or not the technology represented by the model pattern is successful, both in terms of its own goals and the goals of the contextual actors. Compare the results with the evaluation of the existing technology in the contextual model.

Case Study Application. In the case of the wiki pattern, two possible wiki configurations were evaluated, one with periodic reviewing of content and one where content must be reviewed before being posted to the wiki. Although the second configuration proved to be most successful, overall, based on an evaluation of the goals in the contextual model, the wiki technology did not seem to meet the needs of the organization.

7. Improve Pattern & 8. Repeat. Use the experience of inserting and integrating the model patterns to make any necessary adjustments or improvements to the pattern. For each relevant pattern, repeat steps 1 to 7.



	Context. Model (CM)	Wiki Pattern (MP)	Disc. F. Pattern (MP)
Size			
Actors	21	6	4
Elmts.	178	102	76
Links	222	169	96
Total	421	277	176
Context. Model with Integrated Patterns		Wiki Pattern (CIP)	D. F. Pattern (CIP)
Effects of Integration (Steps 4 & 5)			
Elmts.	Add	20	27
	Delete	9	13
	Change	6	0
Links	Add	55	61
	Delete	23	23
	Change	2	7
Actor	Delete	1	1
Overall Resulting Size			
Actor		24	24
Elmts.		272	232
Links		365	321
Total		661	577

Fig. 2. Pattern Methodology and Resulting Measurements

The results of the case study, along with the results of the study described in the following section, are analyzed in Section 7.

6. Exploratory Experiment: Classroom Setting

Although results collected in the case study have potential to address our research questions, the study had several limitations, several of which related to internal validity. First, the modelers who applied the model patterns were often their creators, which is not necessarily the case in pattern-oriented approaches. Second, the evaluation of the pattern approach in the case study was performed by the authors of this work. In order to address some of these limitations, we designed and executed a follow-up experiment in order to find further evidence to address our research questions.

6.1. Exploratory Experiment Context and Preparation

The experiment took place in a graduate course of a school with a focus on business and technology. Thus, the students had a mixed background of technical and business experience. The students had some knowledge of the i* framework through previous courses, but they had not applied it extensively and could be considered to be novice modelers. The study was introduced to the course as one of the course assignments; however, participation was anonymous and voluntary, not affecting grading. Six students opted to participate in the experiment.

The student assignment was divided into three parts. Part A (Contextual Model Creation) simulated the creation of a contextual model, with each student analyzing a type of information technology as applied to a collaborative work setting. All students analyzed the same work setting, but used different technologies. Models were evaluated to explore the effectiveness of the technology. Part B (Pattern Creation) involved the creation of a model pattern, with each student producing a pattern for a technology that they did not choose in the first stage. The third part, Part C (Pattern Integration), required the students to apply and contextualize a selected pattern produced by another student into their model created during Part A. Hints for integration were given to the students by describing some of the steps presented in the methodology in section 4.2. Questions were posed in the assignment to qualitatively assess effort (Q1), coverage (Q2) and complexity (Q3) of various steps in the assignment.

6.3. Qualitative Analysis of Experimental Results

Q1: To address modeling effort, the students were asked which of the assignment activities were the most difficult for them to complete. One student said this was the construction of the Part A (contextual) model, two students indicated that making the Part B pattern was either the most difficult or time consuming to construct, complaining about the difficulty of having to make a more abstract model, and two student said that the integration in Part C was the most time consuming task, with three students complaining about the difficulty of understanding the Part B model. The last student did not clearly pick a task as most difficult.

Q2: To address their perception of coverage, the students were asked the following question, with student answers summarized in Table 1 and the # symbol indicating answers that were missing or unclear.

How would you describe your confidence on the correctness (including accuracy and completeness of coverage) of the models and analysis results of:

1. The Part A model before you performed Part C?
2. The Part A model after you performed Part C?
3. The Part C Model? (in comparison to the Part A model)

Table 1: Summary of Student Answers for Question 2

	Q 2.1.	Q 2.2.	Q 2.3.
Student 1	Correctness: above average Coverage: not sufficient	Accuracy: good Completeness: not as good as part C	#
Student 2	Correctness: High confidence Coverage: cannot be determined	Correctness: High confidence Coverage: cannot be determined	C more complete in coverage
Student 3	Accuracy: good Completeness: good	Completeness: was not as good as thought	Part C model more complete
Student 4	Completeness and Accuracy: not confident (due to lack of i* experience)	Completeness and Accuracy: not confident (due to lack of i* experience) did add more things	Completeness and Accuracy: more confident, but still not completely confident
Student 5	Accuracy: high	Accuracy: not as high as thought	Part C models most accurate and correct
Student 6	Quality and Accuracy: Not confident	Lots of details left out in part A	#

Q3: When asked which models would be the easiest to understand for themselves or for others (related to our research question of model complexity), four students said that the Part C model is easiest for them to understand, while two indicated Part A. However, only two students clearly said that the Part C models would be easiest for others to understand, with one student indicating it would depend on the modeler's experience and another expressing concern about the complexity of Part C.

In addition, when asked about the quality of the model pattern produced by another student, received in Part C, five of the students complained about some aspect of the incoming pattern including completeness, ambiguity and complexity. However, four of these students, as well as the sixth student, listed positive aspects of the incoming pattern including quality and knowledge previously missing.

Finally, despite the concerns expressed, when asked about their overall experience with using patterns in the assignment, five of them said they would use patterns again, although one indicated that only if the pattern was created by a reputable source. The validity of this and other evidence collected is discussed in Section 8.

7. Interpretation and Discussion of Results

In this section, we interpret and discuss the collected evidence from our studies in the light of our three driving research questions.

Q1: Do model patterns help reduce modeling effort? Assuming that patterns are readily available (leaving costs related to pattern construction aside), this research question can be affirmed when the integration of model patterns is less costly than the development of the corresponding parts of non-pattern models.

Case Study: We can acquire an *estimation* of modeling effort by examining the size of this section of the model its sub-agents and related dependencies. By comparing these measures to the amount of effort put into the integration of the discussion forum pattern into the same model, we derive evidence with respect to the question at hand. In our study, the contextually developed discussion forum model has 42 elements and 52 links, compared to the integration of the discussion forum model pattern which required the modification of 40 (13 deletions and 27 additions) elements and 84 links (23 deletions, 61 additions and 7 modifications).

Experiment: To make the experiment's results comparable to our case study results, we would ignore the effort put into the Pattern Creation of the assignment, despite several complaints about the difficulty of this activity. However, apart from pattern construction, there is the act of understanding the pattern sufficiently in order to apply it. As reported, five of the students expressed concerns about their ability to understand the incoming patterns. Furthermore, only one student indicated that Contextual Model Construction was the most difficult to construct while at least three students indicated difficulties with Pattern Integration.

Combined: Examining the Case Study evidence, it appears that the integration of the discussion forum model pattern required at least as much effort as modeling the technology within the contextual model. Considering the experiment, it seems that in addition to problems understanding the incoming pattern, the integration of a model pattern into a contextual model was generally thought to be more difficult than creating the contextualized model. These results are in clear conflict to our predictions, and especially surprising as we have already left the costs related to pattern construction out of the equation. The difference in the level of effort required to integrate patterns between studies may indicate that effort depends heavily on experience, as the case study was performed by experienced i* modelers, while the students in the experiment were i* novices.

Q2: Do model patterns help increase model coverage? This research question can be affirmed when the application of patterns leads to models that cover more *relevant* aspects of the domain than non-pattern models.

Case Study: We have found that patterns have a significant impact in this regard: by replacing the contextually developed discussion forum model with a discussion forum model pattern, model coverage increased along several dimensions: the integration of the pattern introduced 10 additional goals (+143%), 43 additional softgoals (+96%), 49 additional "help" contribution links (+87%) and 14 additional means-ends relationships (+1400%). We can surmise that these were additions of relevant constructs as irrelevant model sections were removed during the contextualization of the model pattern.

Experiment: Even though the students did not have high confidence in the coverage of their contextual models before pattern integration, three students indicated that the integrated models were the most complete and at least two students noticed detail left out of the contextual model after completing the Pattern integration.

Combined: The evidence found in both studies therefore suggests that the adoption of model patterns can have a *positive influence* on *elaborating* i* models with respect to model coverage.

Q3: Do model patterns help decrease complexity? This research question can be affirmed when the application of patterns leads to models which are significantly less complex than non-pattern models.

Case Study: In our case study, all developed model patterns were significantly *more complex* (i.e. contained more elements and links) than their technology counterparts in the contextual models, as discussed in the Q2 analysis. In the example where we have introduced the discussion forum model pattern to replace its contextual counterpart, an overall increase of modeling elements and links of 30% and 43%, respectively, could be observed. Results for replacing the contextual discussion forum model with the wiki pattern showed similar trends. However, the use of patterns can be said to modularize the model development process, and, as the patterns are significantly smaller than the contextual models before and after integration, the complexity of any steps performed with *only* the patterns would be simpler than working with the larger contextual model.

Experiment: We can examine questions relating to the quality of the model pattern and ease of comprehension as measures of model complexity. As mentioned, five of the students expressed concerns about their ability to understand the incoming patterns. In addition, although four of the students indicated that the integrated model would be the easiest to understand, there was concern over the ability of other to understand these models.

Combined: Despite the possible benefits of modularization, as well as the student's purported ability to understand their own integrated models, we are led to doubt an overall reduction in complexity from the use of model patterns. In fact, measuring complexity from model size, the case study results indicate that pattern application may actually increase model complexity.

8. Threats to Validity

Construct Validity: The constructs we intended to investigate in our study were "effort", "model coverage" and "model complexity". In our case study we measured the effort involved in model construction by measuring the amount of necessary model changes (additions, deletions). In doing that, we aimed to eliminate confounding factors such as the varying skills of modelers with a particular modeling tool. However, our approach does not mitigate the potential influence of varying cognitive efforts. In fact, our observations indicate that the act of integrating a pattern into a model may require more cognitive effort than the creation of corresponding, contextual models, which represents an interesting finding.

In the KHP study, we measured model coverage by investigating whether the total amount of modeling elements and links increased or decreased after integration of the model patterns into the contextual model. These changes were made with the relevance of these elements in mind. A potential threat to validity is the subjective nature of "relevance" in general. We tried to mitigate this factor by involving a modeler that has a good understanding of the case study organization. Our case study used the size of the models, including elements and relations, as a measure of model complexity. We argue that this represents a suitable surrogate measure for an exploratory case study. To address issues with the means of measuring effort, coverage and complexity in our case study, our exploratory experiment instead used a qualitative judgment of these aspects as reported by the student participants.

Internal Validity: The internal validity problems of the case study were discussed in Section 6. In the experiment, pattern creation was performed by novice modelers, whereas in pattern theory, patterns are typically developed by experts in the domain and pattern creation. We attempted to mitigate these effects by providing resources on the technology subject matter of the patterns and by providing sufficient i* training. However, results may have differed if the patterns were created by more experienced individuals.

External Validity: Because the experiment and case study were performed using the i* Framework, it is difficult to generalize findings to other modeling frameworks. However, several of our findings may generalize to other agent-oriented, goal modeling frameworks, such as the fact that pattern integration involves significant effort or that patterns have the potential to increase model coverage.

As always, there are external validity issues with the use of students as research subjects, especially when the sample size is small. However, this particular group of students represented a fairly diverse background, having a mixture of academic and business experiences. Furthermore, the subjects had a novice level of expertise in use of the i* Framework, making it difficult to generalize to more experienced modelers. In contrast, the modelers in the case study were experienced with the i* Framework.

Both the case study modelers and the students were experts in their respective domains, KHP and a collaborative work setting. It is possible that differing levels of expertise may produce different findings. However, the issue of expertise in the pattern technology may be yet more relevant, with participants in both studies having varying levels of expertise in the technologies modeled.

The differences in the contexts of our investigations increase our confidence that the results would generalize to other settings. However, it is still possible that some domains may be more amenable to pattern application than others.

The results of our study may depend on the nature of the patterns we use. Employing a variety of pattern creators in both the case study and experiment increases our confidence that the results would generalize to different sized and scoped patterns, but i* patterns defined in a different way may produce different results.

Reliability: Making the methodology we followed explicit increases our confidence that our findings can be reproduced by others. Other than the small number of participants, there is nothing to indicate that, given similar settings, both of our studies would not produce similar results.

9. Related Work

In i* modeling, patterns have not been applied extensively, but some reports are available. [15] and [16], for example, use the i* framework to 1) construct agent-oriented strategic dependency patterns of different types of organizational structures and to 2) (re)construct traditional, object-oriented patterns in an agent-oriented fashion. In addition, [18] uses i* strategic dependency and strategic rationale diagrams to capture and encapsulate knowledge about possible design trade-offs of submarine maneuvering systems for reuse in future engineering efforts. Reusable security patterns, expressed in the i* Framework, are introduced in [17]. While these examples demonstrate the potential of patterns for agent-based, social focused modeling, testing the assumption that a pattern-based approach actually improves modeling was not in the focus of these investigations.

In the broader context of *requirements engineering*, patterns have been proposed and used for many different purposes. Patterns were proposed and investigated as a means for organizing and documenting, for example, functional and non-functional requirements knowledge ([5], [9]) and for capturing knowledge about requirements engineering techniques and strategies [11]. Examples include patterns for refining requirements [6], and dealing with conflicts [19]. Beyond these approaches, patterns were suggested to act as solution templates for requirements specification (Hosoya in [10]), and as guidelines for performing and improving the requirements process [12]. Finally, patterns were investigated as references for assuring the quality of specifications (Hanyuda in [10]). In *software engineering* in general, patterns have a longer tradition. Beyond the influential work on object-oriented patterns (including [3] and [4]), a series of approaches for utilizing *agent-oriented* design patterns have been proposed including [2] and [14].

10. Conclusions

Execution of the studies in this work has revealed some limitations to the use of model patterns in i*. For instance, contrary to our expectations, replacing the technologies in our case study with the two patterns did not have a large effect on the overall goals of KHP's actors. This emphasizes that the application of patterns to a model is a bottom-up (solution driven) approach, whereas the traditional goal modeling approach is predominately top-down (goal driven). Although applying patterns was useful for improving coverage, further brainstorming is required to sufficiently satisfy the goals of the organization.

Execution of the experiment revealed potential difficulties with the construction and comprehension of model patterns. Some students had difficulty constructing patterns capturing abstract situations. Patterns created by other students were often difficult for a student to understand or apply. Further studies should test whether these issues are as apparent when models are created and used by experienced modelers.

Can patterns improve i* modeling?

Q1-Effort: We have found that several assumptions of pattern theory seem to be questionable when applied to i* modeling. Even when we took the effort necessary for pattern creation out of the equation, we found empirical evidence that suggests that patterns increase modeling effort for novice users, and do not decrease effort for more experienced users.

Q2-Coverage: The findings of our exploratory investigations suggest that the utilization of patterns can address issues identified with i* modeling related to coverage by integrating broader domain knowledge.

Q3-Complexity: We could not find evidence that patterns help in reducing complexity in an i* context. In fact, our quantitative case study findings suggest the opposite: pattern integration almost always led to an increase of modeling elements. Our qualitative experimental findings also point to an increase in the complexity of models containing patterns, especially for those not creating the models. However, because patterns also modularize the domain and can be inspected independent from their contexts, patterns might nevertheless support analysts in dealing with large-scale models. Further studies should investigate this possibility.

Combining these preliminary observations, we can make the assertion that the decision to apply patterns in a given situation can be made based on certain factors including the importance of model coverage and the experience of the modelers. If model coverage, including related factors such as accuracy and correctness, are strongly desired, applying a tested and reputable pattern can be beneficial, especially if being applied by experienced modelers. However, if reduced effort and complexity are favored over coverage, or if modelers are inexperienced, a pattern approach may be less appropriate.

In this paper, we have investigated the application of model patterns in the presence of existing contextual models. One promising further application is the utilization of patterns at the beginning of the modeling process, where contextual models have not yet been created. In this situation, model patterns could be used as seeding elements for the construction of contextual models, eliminating the effort of pattern integration. Relevant topics for future research brought to light by our exploratory studies include examining the impact of patterns on model comprehension and correctness, as well as further investigating the effect of modeler experience and domain expertise on the ability to effectively apply patterns.

References

1. Christopher Alexander. The Timeless Way of Building. Oxford Press, 1979.
2. Y. Aridor and D.B. Lange. Agent design patterns: elements of agent application design. In Proceedings of the second international conference on Autonomous agents, pages 108–115. ACM Press New York, NY, USA, 1998.
3. K. Beck, J. O. Coplien, R. Crocker, L. Dominick, G. Meszaros, F. Paulisch, and J. Vlissides. Industrial Experience with Design Patterns. In Proceedings 18th International Conference on Software Engineering (ICSE), 1996.
4. F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal. Pattern-Oriented Software Architecture, Volume 1: A System of Patterns. Wiley, 1996.
5. L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos. Non-Functional Requirements in Software Engineering. Kluwer Academic Publishers, 2000.
6. R. Darimont and A. van Lamsweerde. Formal refinement patterns for goal-driven requirements elaboration. ACM SIGSOFT Software Engineering Notes, 21(6):179–190, 1996.
7. S. M. Easterbrook, E. Yu, J. Aranda, Y. Fan, J. Horkoff, M. Leica, and R.A. Qadir. Do Viewpoints Lead to Better Conceptual Models? An Exploratory Case Study. In Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE’05), pages 199–208, 2005.

8. E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns - Elements of Reusable Object-Oriented Software. Addison Wesley, 1995.
9. D. Gross and E. Yu. From Non-Functional Requirements to Design through Patterns. Req. Eng., 6(1):18–36, 2001.
10. L. Hagge, F. Houdek, and B. Paech. Workshop Summary of the International Workshop on Requirements Engineering Patterns, In conjunction with RE’04, the 12th IEEE International Requirements Engineering Conference, 2004.
11. L. Hagge and K. Lappe. Sharing requirements engineering experience using patterns. IEEE Software, 22(1):24–31, 2005.
12. L. Hagge and K. Lappe. Using Requirements Engineering (RE) Patterns for Organizational Learning. Journal of Universal Knowledge Management, 1(2):137–148, 2006.
13. J. Horkoff, E. Yu, L. Liu. (2006) Analyzing Trust in Technology Strategies. International Conference on Privacy, Security and Trust (PST 2006), pages 21-32, 2006.
14. E.A. Kendall, P.V.M. Krishna, C.V. Pathak, and CB Suresh. Patterns of intelligent and mobile agents. In Proceedings of the second international conference on Autonomous agents, pages 92–99. ACM Press New York, NY, USA, 1998.
15. M. Kolp, P. Giorgini, and J. Mylopoulos. Organizational Patterns for Early Requirements Analysis. In Fifth IEEE International Symposium on Requirements Engineering (RE01). Springer, 2001.
16. M. Kolp, P. Giorgini, and J. Mylopoulos. A Goal-Based Organizational Perspective on Multi-Agent Architectures, pages 128–140. Springer, 2002.
17. H. Mouratidis, P. Giorgini, and M. Schumacher. Security Patterns for Agent Systems, In Proc. of Eighth European Conf. on Pattern Languages of Programs, pages 25- 29, 2003.
18. P. Pavan, N.A.M. Maiden, and X. Zhu. Towards a Systems Engineering Pattern Language: Applying i* to Model Requirements-Architecture Patterns. In Proceedings of the Second International Workshop From Software Requirements to Architectures, in conjunction with ICSE, 2003.
19. A. van Lamsweerde, R. Darimont, and E. Letier. Managing conflicts in goal-driven requirements engineering. IEEE Transactions on Software Eng., 24(11):908–926, 1998.
20. E. Yu. Modelling Strategic Relationships for Process Reengineering. PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada, 1995.
21. E. Yu. Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. Proceedings of the 3rd IEEE Int. Symp. on Requirements Engineering (RE'97), USA, pp. 226-235, 1997.