Requirements Reuse and Patterns: A Survey

Cristina Palomares, Xavier Franch, Carme Quer

GESSI Research Group, Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
{cpalomares, cquer, franch}@essi.upc.edu

Abstract. *Context and motivation:* Multiple proposals exist that propose the adoption of reuse practices during requirements engineering processes. *Question/problem:* Which is the current level of adoption of these practices in organizations? *Principal ideas/results:* In this paper we present the preliminary results of a survey initiated at REFSQ'13 that addresses this question. The survey first investigates requirements reuse in general, and then goes in depth asking about a specific technique, software requirement patterns (SRP), which is the backbone of our PABRE framework. *Contribution*: The survey results show that requirements reuse is not a widespread practice in IT projects, being the most common techniques those based on the copy and later modification by hand of requirements coming from previous projects. Regarding the use of SRP, the results seem to support our hypothesis that SRP could help to ameliorate some common problems related to requirements specifications like lack of uniformity and incompleteness.

Keywords: Requirement engineering; Requirement reuse; Requirement Patterns.

1 Introduction

The PABRE (PAttern-Based Requirements Elicitation) framework is the result of the collaboration between the GESSI research group at the UPC and the SSI department at the Public Research Centre Henri Tudor (TUDOR) in Luxembourg to adopt software requirement patterns (SRP) as an approach to reuse. PABRE includes a metamodel for SRP [1], a catalogue of 45 Functional SRP (for the content management system domain) [2], 29 Non-Functional SRP [3] and 37 Non-Technical SRP [4] (best suited for business information systems like customer relationship management, supply chain management and by the like), and several tools for SRP management and use [5].

The formulation of the framework heavily relies on empirical work. The collaboration with TUDOR made it possible to analyze requirement specification documents used in industrial projects, which at its turn required a thorough systematic literature review on requirements reuse in general, and SRP in particular. As part of this empirical approach, we decided to investigate the current perception of requirements reuse by practitioners and academics and that's why we designed and conducted a survey (http://www.upc.edu/gessi/PABRE/Survey.html). The goal of the paper is to report preliminary results on the current use of requirements engineering reuse practices in organizations and their benefits and drawbacks and, taking into account our specific approach to requirements reuse, to know the opinion of participants about SRP as reuse artifact. It is worth to remark that the current paper focuses then in the state of the practice, not consid-

ering the responses that come from researchers without industrial experience which remains subject of later analysis.

2 Research Method

Research questions.
- *RQ1: Is requirements reuse a usual practice in current RE processes?* Here we investigate the current situation of requirements reuse practices in organizations, i.e. the level of requirements reuse, the type of requirements that are more prone to be reused and the techniques used to achieve it.
- *RQ2: Which benefits and drawbacks can appear from the use of a catalogue of SRP?* Taking into account our specific field of research, we are especially interested in SRP as reuse artefact. This is the reason for asking to the participants in which degree requirements engineering problems can be ameliorated by the existence of an SRP catalogue and about critical aspects and barriers for its introduction in an organization.

Data points. The data points considered are practitioners and academics with either "significant" or "some" level of experience on requirement engineering. We adapted the questions to each level of experience in order to obtain more accurate responses.

Channel. The survey is implemented as an online questionnaire. Firstly, it was offered to REFSQ 2013 assistants (as part of the Empirical Track) who could answer it during the conference. Afterwards it was also offered to the requirements engineering community through other channels as LinkedIn requirement engineering groups, related tutorials attendees in conferences as RE 2013 and ICSE 2013, and online communities.

Data analysis. The results presented in this paper are based in descriptive statistics and content analysis (the last one only for questions with results in free text). In case of the questions using Likert scales, their results have been analyzed following the good practices presented in [6]. We are waiting to increase the number of data points before performing correlation and cluster analysis.

Questionnaire design. In order to avoid typical design errors in online surveys, we accompanied critical questions with a glossary of terms; we added whenever necessary text fields for clarification or for allowing the respondent adding missing values; and we conducted pilots of the questionnaire to ensure its correct understanding and its possible display effects. An excerpt of the online survey containing the questions necessary to answer the research questions analyzed in this paper can be found in [7].

Validity. On the one hand, in order to have a random sampling, aside from proposing the survey in several conferences, we introduced it as discussion topic in the main LinkedIn requirement engineering groups, and we introduced it as an open discussion in the groups to engage not only people that are already using requirement reuse practices. On the other hand, in order to mitigate the problem of coverage of the requirements engineering population and the low response-rate common in online surveys, we proposed the survey through the LinkedIn and community groups with around 12000 members altogether.

3 Results and Discussion

At the moment of writing the paper, we had 50 completed responses from practitioners and researchers with industrial experience, from 19 countries around the world (mostly from North America and Europe). From them, 27 (54%) were requirement engineers in industry, 10 (20%) researchers with significant experience as requirement engineers, and 13 (26%) researchers with some limited experience as requirement engineers.

RQ1: Is requirements reuse a usual practice in current RE processes?

We asked participants about three different aspects (see *About Reuse during Requirements Engineering* and *About Observations on Requirements* sections in [7]):

- *What is the level of requirements reuse they had in their projects* (see Table 1).
- *Which are the types of non-functional and non-technical requirements (NFR, NTR respectively) that were more similar from project to project*. A list of NFR and NTR types was provided so the similarity of each type among projects could be measured (see Table 2). It was possible to add other NFR or NTR types that could be relevant and not stated by the survey.
- *What are the techniques they implemented to achieve requirements reuse* (multiresponse question). This last question was asked only to those participants that implemented some kind of requirements reuse in their projects, i.e. being the requirements reuse level in the first question marked at least as *2-Low* (see Table 3).

Table 1: Requirements reuse level

	#Participants (%Participants)
Not able to answer	1 (2%)
Inexistent or Very Low	11 (22%)
Low	18 (36%)
Medium	9 (18%)
High	9 (18%)
Very High	2 (4%)
TOTAL	*50 (100%)*

Table 3: Requirements reuse techniques

	#Participants (%Participants)
Copy & Paste of Individual reqs.	23 (60%)
Copy & Paste of Groups of reqs.	21 (55%)
Duplicate of a full reqs. specification	19 (50%)
Fill in predefined templates	15 (39%)
Use of a req. patterns catalogue	5 (13%)

Table 2: Requirement types more similar between projects (1 – Totally disagree, 5 – Totally agree)

	Likert Scale Average
4. Usability	3,60
5. Reliability	3,48
6. Security	3,44
7. Maintainability	3,44
2. Performance Efficiency	3,24
11. Business Suitability	3,20
12. Project Suitability	3,16
3. Compatibility	3,08
8. Portability	3,08
1. Functionality Suitability	3,04
10. Product Non-Technical Suitability	2,94
9. Supplier Suitability	2,92

Regarding the first question (Table 1), we got 38 participants (76%) that stated the requirements reuse level as equal or greater than *2-Low*. However, reuse seems not to be an established practice in IT projects since only 22% of the participants marked it as equal or greater than *4-High*.

The results of the second question about the types of NFR and NTR that were more similar from project to project (Table 2) do not highlight a big difference in the level of recurrence of the types, being most of them around 3 (equivalent to *Neutral* value). One of

the possible reasons for this neutrality is that most of the people that took part in the survey did not carry out requirements reuse. The four requirement types that were ranked with a higher reuse rate were: Usability, Reliability, Security, and Maintainability.

Regarding the techniques used to implement requirements reuse, the current results of the survey (Table 3) show that the most common techniques, used by more than 50% of the participants, are those based on the textual copy and later modification by hand of requirements coming from previous projects, i.e. *Copy and paste of individual existing requirements* or *Copy and paste of groups of requirements* in the requirements specification under construction and *Duplicate of a full existing requirements specification* and work in its parts as needed. Less common techniques seem to be *Fill in of predefined templates* and *Use of a requirement patterns catalogue*.

Discussion

IT practitioners include reuse in their daily practices, although the reuse process is probably most of the times simple Copy & Paste with its corresponding problems. The rationale behind is simple: during elicitation and definition of requirements, whenever the IT professional remembers some previous project where requirements looked close to the ones of the current project, copying and using them as a starting point seems natural. We think this is the reason why we found that 76% of participants declare to carry out some level of reuse in their projects. Thus our interpretation of the results is that requirements reuse is present in IT projects, but well-defined and mature reuse methods and processes have still to emerge and be integrated into the practice of organizations.

Concerning the type of requirements more similar among projects, non-functional requirements (types numbered from 2 to 8) are considered as more similar among projects than functional ones (numbered as 1), which is corroborated by the fact that non-functional requirements is the main focus of requirements reuse proposals [8][9][10]. In case of non-technical requirements (numbered from 9 to 12 most), the results are not the ones we expected. For instance, in case of the *Supplier Suitability,* which was defined in the questionnaire as corresponding to those requirements on the organization that supplies the software product, it was considered less recurrent than functional requirements. Our interpretation is that non-technical requirements were not well understood by participants, since according to our experience, this kind of requirement is in fact quite recurrent, if we think for instance on requirements about the positioning and strength of the supplier organization, the certifications that this organization has on software processes development, the services it offers or its maintenance and development procedures. This misunderstanding could be caused by the fact that these kinds of requirements are not always included in requirement specifications unless projects are call-for-tenders projects.

Chernak conducted an online survey on requirements reuse during 2010 [11] (henceforth *CheS*). One of its questions can be directly related to the first aspect addressed in RQ1. Its results were that 59% of respondents reused requirements in their latest projects. If we compare this magnitude with ours, in our survey the percentage is higher: considering the respondents with a level of reuse equal or above Low, we observe that 76% of interviewees do some reuse in their projects (Table 1). This difference may be caused by the different population of the surveys: CheS involved software engineers in general, whilst in our case we addressed requirements engineering practitioners. In addition, CheS survey's results indicate that practitioners that adopt reuse practices do not usually follow well-defined reuse processes, sustaining also our results.

RQ2: Which benefits and drawbacks can appear from the use of a SRP catalogue?

To answer this research question we asked the participants to evaluate, using Likert Scales, a list of problems that could be ameliorated by the use of a SRP catalogue, and two lists of critical factors and barriers respectively that could affect the successful adoption of an SRP catalogue (see *About Reuse through Patterns* section in [7]). In the three lists, the participants had the opportunity to add new items not appearing in the lists.

Regarding the problems that could be ameliorated by the use of an SRP catalogue (see Table 4), the four most mentioned problems in the survey are: *Lack of requirements uniformity*, *Incompleteness of requirements specification*, *Ambiguity of requirements* and *Having too little time to spend in requirements elicitation*.

Regarding the factors that could be critical for the introduction of a SRP catalogue (see Table 5), all the listed factors were considered critical except the existence of *Help Desk*. Participants added other critical factors, being the most mentioned ones *The existence of a ready-to-use SRP catalogue*, *The possibility of having free trials periods*, and *The existence of success cases using SRPs*.

Table 4: Problems ameliorated by the use of a SRP catalogue (1 – At all, 3 - A lot)

	Likert Scale Average
Lack of requirements uniformity	2,43
Incompleteness of requirements specification	2,37
Requirements ambiguity	2,32
Too little time invested in requirements elicitation	2,31
Requirements non-verifiable	2,21
Too much time spent in requirements elicitation	2,20
Stakeholders do not know exactly their needs	2,18
Stakeholders needs' change during the requirements elicitation process	2,18
Requirements inconsistency	2,16
Lack of requirements. traceability	2,12
Lack of requirements quantification	2,12
Lack of requirements prioritization	2,04
Conflicts among needs stated by stakeholders	2,00

Table 5: Critical factors for introducing a SRP catalogue (1 – Totally disagree, 5 – Totally agree)

	Likert Scale Average
Well-defined use method	4,22
Tool support	4,12
Community of users existence	3,94
Training courses	3,92
Help desk	3,37

Table 6: Barriers to adopt successfully a SRP catalogue (1 – Totally disagree, 5 – Totally agree)

	Likert Scale Average
Resistance of req. engineers to change	4,10
Integration of the catalogue with the existing req. engineering processes	3,96
Risk of converting requirements elicitation in a stiff process	3,63
Amount of reusable knowledge necessary to create and maintain	3,59

Finally, for the list of factors that may represent a barrier to the successful adoption of an SRP catalogue (see Table 6), all of its items were considered as important barriers that should be taken into account, being the highest rated ones: *The resistance of requirements engineers to change*, and *The integration of the catalogue with the existing requirements engineering process*. Another barrier not included in the list but considered as very important by some participants was *The lack of management support*.

Discussion

The results of the first question are very important because the three problems identified by the survey participants corroborate the aim of our PABRE framework: to increase the uniformity and completeness of requirement specifications and to reduce ambiguity among requirements in these specifications.

The critical factors obtained as more relevant also support our interpretation of the RQ1 results. The importance given to the existence of a reuse method and tool sup-

port is probably caused by the absence of a well-defined and mature method to guide the reuse processes undertaken by the participants. Regarding the barriers, it is not surprising that the ones that depend on people involvement are considered the most important: when we talk about processes in organizations, the implication of involved people become a key factor for the adoption and its success [12].

In the CheS survey [11] there are two questions that can be considered indirectly related to RQ2, which ask about benefits and obstacles. In our case, we ask about similar aspects but specifically for reuse through patterns. The main benefit reported by the CheS survey participants was "faster-time-to-market", mentioned by 50% of their respondents. We also had this response (*Too much time spent in requirements elicitation* in Table 4) as a possible answer of problems that could be ameliorated by the use of a SRP catalogue. Although in our case it was not ranked as the most important benefit, it was still considered as relevant by the participants. This "faster-time-to-market" benefit has been proved to be true in real processes, such as Goldin et al. case study [13] conducted in an organization that was incorporating requirements reuse to their requirements engineering processes. On the other hand, CheS' obstacles can be assimilated to barriers in our survey. Both surveys include as barriers the creation and maintenance of reusable artifacts, and the resistance of project managers and requirement engineers (see Table 6). It is difficult to compare both surveys because in CheS, the answers to these questions were open, and in ours a list was provided with an open field to extend it if needed; however, we still may observe that CheS' results are included in ours.

In Hoffmann et al. [14], the result of 5 semi-structured interviews (henceforth *HKHL*) with experienced requirements analysts is presented. They were asked on their opinions on the advantages and success factors they could perceive on the use of SRP approaches. In the case of advantages, 7 out of the 8 most important problems identified as potentially ameliorated by the use of SRP in our survey (Table 4) were also stated in the HKHL interviews. The only one that did not appear in HKHL was *Too little time invested in requirements elicitation*. In the case of success factors and barriers, all of those identified in our survey were also found in HKHL, and the ones in HKHL that were not in our survey are more related to the quality of the SRP catalogue, which we did not considered as an option because we gave it for granted.

4 Conclusions and Further Work

In this paper, we presented the preliminary results of a survey to study the state of the practice on requirements reuse, and on the possible advantages, success factors and barriers of using SRP as reuse artifact. The survey results show that requirements reuse, although is not a widespread practice in IT projects, is used in a certain level in the projects were survey participants participated, but probably there is a lack of well-defined and mature reuse methods and processes. Regarding the use of SRP, the results seem to support our hypothesis that SRP could help to ameliorate some common problems related to requirements specifications like lack of uniformity, incompleteness and ambiguity. The aspects more critical in the application are the implication of

requirement engineers and project managers and also the existence of a well-defined method of use and the existence of tool support.

Future work includes the extension of this study by gathering more responses, to include also responses given by researchers without practical experience and conducting correlation analysis and cluster analysis of responses considering also other general questions included in the questionnaire.

Acknowledgements

References

1. Franch, X., Palomares, C., Quer, C., Renault, S., De Lazzer, F.: A Metamodel for Software Requirement Patterns. In: Int. Work. Conf. on Requirements Engineering: Foundation for Software Quality (REFSQ), pp 85--90. Springer (2010).
2. Palomares C., Quer C., Franch X., Guerlain C., Renault S.: A Catalogue of Functional Software Requirement Patterns for the Domain of Content Management Systems. In: Requirements Engineering Track at 28th ACM SAC (RE-SAC), pp 1260-1265. ACM (2013).
3. Renault, S., Mendez, O., Franch, X., Quer, C.: A Pattern-based Method for building Requirements Documents in Call-for-tender Processes. In: Int. J. of Computer Science & Applications. 6(5), pp 175-202. TMRF (2009).
4. Franch, X., Quer, C., Renault, S., Guerlain, C., Palomares, C.: Constructing and Using Software Requirements Patterns. In: Managing Requirements Knowledge, Maalej, W., Thurimella, A.K. (Eds.), pp 95-116. Springer (2013).
5. Palomares, C., Quer, C., Franch, X.: PABRE-Proj: Applying Patterns in Requirements Elicitation. In: IEEE International Requirement Engineering Conference (RE), pp332-333. (2013).
6. Jamieson, S.: Likert Scales: How to (Ab)use Them. In: Medical Education, Vol. 38(12), pp.1217-1218 (2004).
7. Report of the *Requirements Reuse and Patterns* survey - excerpt. http://www.upc.edu/gessi/PABRE/Survey_Questions.pdf
8. Jaramillo A.F. Non-functional requirements elicitation from business process models. 5th International Conference on Research Challenges in Information Science (RCIS). IEEE CS Press (2011).
9. Hoffmann, A., Schulz, T., Hoffmann, H., Jandt, S.. Roßnagel, A., Leimeister, J.M. Towards the Use of Software Requirement Patterns for Legal Requirements. 2nd International Requirements Engineering Efficiency Workshop (REEW) at REFSQ. (2012).
10. Toval A., Carrillo-de-Gea J.M., Fernandez-Aleman J.L., Toval R., Learning systems development using reusable standard-based requirements catalogs. Global Engineering Education Conference (EDUCON). IEEE CS Press (2011).
11. Chernak, Y.: Requirements Reuse: The State of the Practice. In: Software Science, Technology and Engineering (SWSTE), pp.46--53. IEEE CS Press (2012).
12. Dyba, T. "An empirical investigation of the key factors for success in software process improvement." Software Engineering, IEEE Transactions on 31(5), IEEE CS Press (2005).
13. Goldin, L., Berry, D.M.: Reuse of requirements reduced time to market at one industrial shop: a case study. In: Requirements Engineering Journal. 18. Springer (2013).
14. Hoffmann, A., Janzen, A., Hoffmann, H., Leimeister, J.M.: Success Factors for Requirement Patterns Approaches. In: Sozio-technisches Systemdesign im Zeitalter des Ubiquitous Computing (SUBICO) im Rahmen der Informatik, Koblenz (2013).