Evaluating the Usefulness and Ease of Use of a Software Process Line Tool

Diogo Matheus Costa[1], Eldânae Nogueira Teixeira[1] and Cláudia Maria Lima Werner[1]

[1] Universidade Federal do Rio de Janeiro/UFRJ - COPPE, Rio de Janeiro/RJ, Brasil
{dmc, danny, werner}@cos.ufrj.br

Abstract. Software Process Line (SPrL) provides a systematic way to support software process reuse. However, adopting SPrL approaches without supporting tools may be a very difficult task. Although SPrL supporting tools are becoming more practical, integrated tools are still needed to support all reuse stages. Hence, aiming to support SPrL activities, the Odyssey tool, which provides a software reuse environment, has been improved. Since every proposed technology should be evaluated before being available for use, this paper aims to characterize Odyssey acceptance through its usefulness and ease of use under the perspective of Brazilian software process experts during software process definition using SPrL. This study was designed using Technology Acceptance Model (TAM) and Goal, Question, Metric (GQM) paradigm with a predefined interpretation model. It was performed with seven experts and the collected data provides insights that Odyssey indeed helps the software process definition using SPrL. Although the results are positive, the identified threats highlight the study limitations. Finally, a set of improvements was identified based on the collected data, participants' comments, as well as researchers' notes. These improvements are being implemented.

Keywords: Software Process Line, Supporting Tool, Evaluation.

1 Introduction

Considering the fact that the process by which software is developed influences the final product quality [19, 10], companies are concerned with their process definitions. However, it is considered a complex activity that requires experience and knowledge from a variety of Software Engineering disciplines [1, 2]. Besides that, defining a software process from scratch is risky and requires much time and effort [26, 28].

In this scenario, the Software Process Line (SPrL) technique provides a systematic way to reuse software processes, identifying their similarity and variability. SPrL represents a set of processes in a particular area, or for a specific purpose, that shares characteristics, being built upon reusable process assets [26]. Some of SPrL's expected benefits are: (i) increase of the reuse potential; (ii) risk reduction; and (iii) increase of the processes quality and suitability [20, 12].

However, adopting SPrL approaches without supporting tools may be a very difficult task [3]. The definition of supporting tools can enable the practical use of process reuse approaches, providing a major possibility of their application in real scenarios [25]. Although SPrL supporting tools are becoming more practical, integrated tools are still needed to support all reuse stages [5, 25].

Although there are other tools [1, 3, 14, 15, 22], this work investigates the Odyssey tool [18], which provides a software reuse environment. Odyssey was chosen because of its availability, as well as the improvements made over the past few years to provide integrated support to SPrL concepts and engineering stages [24, 6].

This paper aims to characterize Odyssey acceptance through its usefulness and ease of use under the perspective of Brazilian software process experts during software process definition using SPrL. To achieve this goal an evaluation was designed and performed using Technology Acceptance Model (TAM) [8] and Goal, Question, Metric (GQM) [23].

The remainder of the paper is structured as follows: Section 2 provides some background on SPrL engineering and TAM; Section 3 briefly describes the Odyssey tool; Section 4 details the study design and presents the evaluation results; and Section 5 presents the final considerations, as well as proposals for future work.

2 Theoretical Background

2.1 SPrL Engineering

In its engineering viewpoint, SPrL is divided into two stages [20]: I. Software Process Domain Engineering (SPDE); and II. Project-Specific Process Engineering (PSPE).

In the SPDE stage, development *for* reuse, process engineers define the SPrL reusable artifacts. SPDE activities are focused on understanding the domain, generating a set of models, such as: a) feature model, used to represent the domain knowledge, identifying similarity and variability; and b) context model, used to characterize entities, information and context situations that support the selection of reusable process elements.

In the PSPE stage, development *by* reuse, reusable process elements are selected for a project-specific software process considering the process definition context (e.g., information related to customer, organization, product). PSPE activities are focused on software process definition using SPrL. These activities are: 1) project characterization, identifying its requirements; 2) reusable process elements selection, solving configuration points (i.e., variability); and 3) remaining software process adaptation, to meet specific needs that may not be contemplated in the process domain. The result is a project-specific software process defined with all the elements that should compose its execution.

2.2 Technology Acceptance Model (TAM)

Proposed by Fred Davis [7], Technology Acceptance Model (TAM) was idealized based on Theory of Reasoned Action (TRA), a psychological theory that seeks to explain behavior [9]. Davis concludes that the concepts of perceived usefulness and perceived ease of use influence people's decision to use or not an information system

[7]. In this context, a positive perception of these concepts provides insights of use intent. These concepts are described below:

- **Perceived Usefulness:** The degree to which an individual believes that using a particular system would increase his or her job performance;
- **Perceived Ease of Use:** The degree to which an individual believes that the use of a particular system would be free of physical and mental effort.

3 Odyssey Tool

Odyssey (Fig. 1) is an infrastructure based on domain models, which provides support to development *for* reuse and development *by* reuse [17]. Although this work focuses on the SPrL support, Odyssey also supports other scenarios (e.g., Software Product Line).

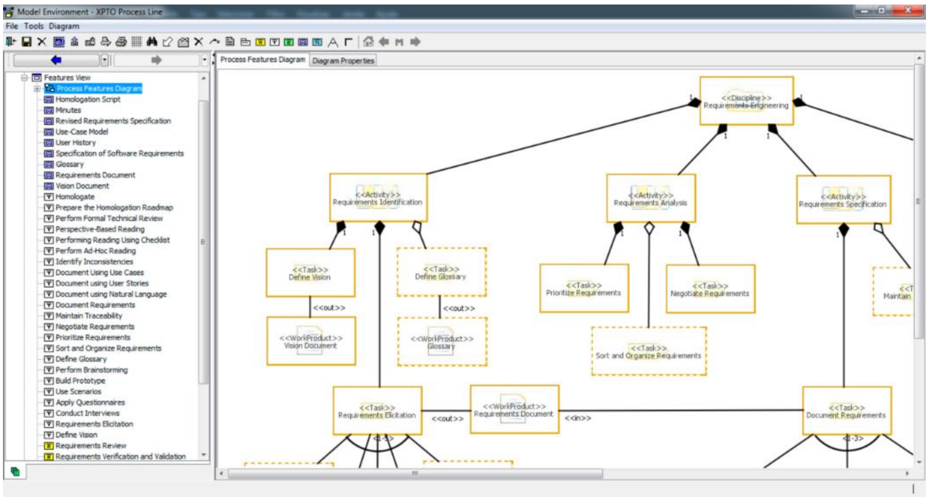

Fig. 1. Odyssey modeling interface.

In the software process context, Odyssey provides support to SPrL engineering activities. Over the past few years, Odyssey has been improved mainly focused on SPDE stage [24]. However, the most recent contributions have focused on PSPE stage support and its integration with SPDE stage [6].

The evaluation presented in this paper focuses on the software process definition using SPrL. In this scenario, Odyssey provides three supporting mechanisms:

- **Slicing Mechanism:** In this mechanism (Fig. 2), the user is supported during the manual selection of process elements through variability and optionality resolution checks, as well as composition rules (i.e., constraints). In this scenario, only the mandatory process elements are pre-selected. During the feature model slicing, selections verifications are conducted and composition rules are analyzed (i.e., Rule-Based System technique) to influence the reusable process elements selection through the defined constraints. If adopted alone, this mechanism provides limited decision-making support. The similarity search mechanism functionalities are available aiming to support the remaining SPrL variability resolution.

- **Context Recognition Mechanism:** In this mechanism, the user describes the software project context by assigning specific values to the context information of the SPrL context model. The support occurs through context rules (i.e., Rule-Based System technique) specified in the SPrL context model based on expected context situations (i.e., actions for expected context definitions). As a result process elements are suggested (i.e., pre-selected) to support the SPrL variability resolution. The functionalities of the slicing mechanism are available, as well as the similarity search mechanism functionalities aiming to support the remaining SPrL variability resolution.
- **Similarity Search Mechanism:** In this mechanism, the user describes the software project context through context entities and information. Based on this characterization, a similarity analysis (i.e., Case-Based Reasoning technique) is performed on the software processes related to the SPrL. The result of this similarity analysis is a list of software processes defined in a similar context, their basic information, representation model and context information.

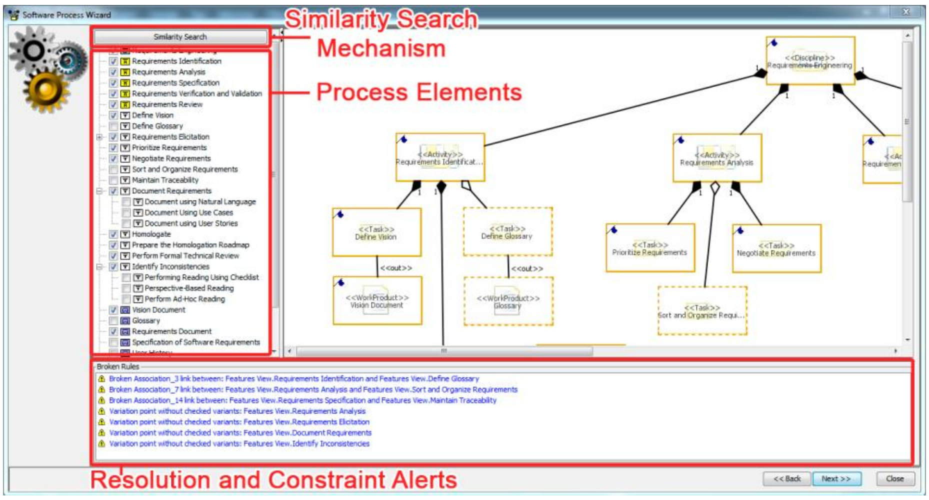

Fig. 2. Odyssey slicing mechanism interface.

The use of the similarity search mechanism alone aims at the full project-specific software process reuse. The goal of full reuse is to adopt a software process defined in a similar context as a starting point to define a project-specific software process. On the other hand, the use of the similarity search mechanism through other mechanisms (slicing mechanism and context recognition mechanism) aims at the partial project-specific software process reuse. The goal of partial reuse is to analyze the decisions taken in software processes defined in a similar context to support the variability resolution.

The Odyssey's software process definition mechanisms are presented in [6], through a usage scenario with illustrations.

4 Evaluation

As mentioned before, Odyssey has evolved over the past few years to support SPrL engineering activities. However, any body of knowledge must be evaluated to be considered scientific [13]. In addition, according to BASILI et al. [4], all proposed technology (e.g., tool, method) must be evaluated before being available for use.

Thus, this evaluation aims to characterize Odyssey acceptance. Besides that, this work can be classified as an observation study, where the participant performs tasks while being observed by an experimenter [21].

In this scenario, a plan was defined based on TAM. In addition, the GQM paradigm was used and the items proposed by DAVIS [8], focused on perceived usefulness and perceived ease of use are applied as questions to characterize acceptance and predict systems usage. Finally, to capture the participant answer and assess each question, the Likert psychometric scale [16] was used to determine the participants' level of agreement or disagreement.

4.1 Planning and Definition

Objective. *Analyze* the Odyssey tool; *with the purpose of* characterizing its acceptance; *with respect to* the perceived usefulness and perceived ease of use; *from the point of view of* Brazilian software process experts; *in the context of* software process definition using SPrL's artifacts defined by the researcher.

This objective does not involve any kind of comparison with other tools. In addition, this work is not intended to answer any performance-related questions to evaluate which is the best software process definition mechanism (Section 3).

Goals, Questions and Metrics. A model was built based on the GQM paradigm, composed of five specific goals, fifteen questions and five metrics. The specific goals are presented in Table 1.

Table 1. Evaluation specific goals.

Goal	Description
G1	Evaluate the training material appropriateness.
G2	Evaluate the similarity search mechanism support to the concept of full project-specific software process reuse.
G3	Evaluate the similarity search mechanism support to the concept of partial project-specific software process reuse.
G4	Evaluate the perceived usefulness of the Odyssey tool.
G5	Evaluate the perceived ease of use of the Odyssey tool.

The G1 goal is focused on the training material, G2-G3 are focused on the support offered by the similarity search mechanism, because it is the Odyssey most recent contribution, and G4-G5 represent TAM's goals (i.e., perceived usefulness and perceived ease of use).

To analyze these specific goals (Table 1), fifteen questions were defined, which are presented in Table 2. As mentioned before, the items proposed by DAVIS [8], focused on perceived usefulness and perceived ease of use, are applied as questions (Q4-Q15), representing the evaluation main questions.

Table 2. Questions applied to analyze the specific goals.

Goal	Question	Description
G1	Q1	The training material applied was sufficient to perform the tasks.
G2	Q2	The similarity search mechanism provides satisfactory support to reuse software processes defined in a similar context as a starting point to adapt them aiming at the project-specific software process definition.
G3	Q3	The similarity search mechanism provides satisfactory support to visualize the decisions made in software processes defined in a similar context aiming at the SPrL variability resolution support during project-specific software process definition using the slicing mechanism.
G4	Q4	Using Odyssey in my job would enable me to accomplish tasks more quickly.
G4	Q5	Using Odyssey would improve my job performance.
G4	Q6	Using Odyssey in my job would increase my productivity.
G4	Q7	Using Odyssey would enhance my effectiveness on the job.
G4	Q8	Using Odyssey would make it easier to do my job.
G4	Q9	I would find Odyssey useful in my job.
G5	Q10	Learning to operate Odyssey would be easy for me.
G5	Q11	I would find it easy to get Odyssey to do what I want to do.
G5	Q12	My interaction with Odyssey would be clear and understandable.
G5	Q13	I would find Odyssey flexible to interact with.
G5	Q14	It would be easy for me to become skillful at using Odyssey.
G5	Q15	I would find Odyssey easy to use.

Finally, each question is answered using a Likert scale composed of five alternatives (i.e., five points) and is assessed by the metrics presented in Table 3.

Table 3. Metrics applied to assess the questions.

Metric	Description
M1	Number of participants who chose “Strongly Disagree”.
M2	Number of participants who chose “Partially Disagree”.
M3	Number of participants who chose “Indifferent”.
M4	Number of participants who chose “Partially Agree”.
M5	Number of participants who chose “Strongly Agree”.

Interpretation Model. To analyze the collected data, an interpretation model was defined based on the specific goals, its questions and metrics. The model is presented in Table 4 and it should be interpreted by checking each expression to confirm its veracity, following the format "IF *expression*, THEN *interpretation*". In this scenario, the interpretation model was defined based on the evaluation conducted in the work of HERNANDES et al. [11].

Table 4. Data interpretation model.

Nº	Question	Expression	Interpretation
1	Q1	M5+M4+M3 >= M2+M1	The provided training material was sufficient to conduct the evaluation.
2	Q1	M5+M4+M3 < M2+M1	The provided training material was insufficient and may have influenced the result.
3	Q2	M5+M4 >= M3+M2+M1	The similarity search mechanism support was sufficient to meet the concept of full reuse.
4	Q2	M5+M4 < M3+M2+M1	The similarity search mechanism support was insufficient to meet the concept of full reuse. Therefore, participants' comments should be analyzed to identify improvements.
5	Q3	M5+M4 >= M3+M2+M1	The similarity search mechanism support was sufficient to meet the concept of partial reuse.
6	Q3	M5+M4 < M3+M2+M1	The similarity search mechanism support was insufficient to meet the concept of partial reuse. Therefore, participants' comments should be analyzed to identify improvements.
7	Qi, i=4 to 9	M5+M4 >= M3+M2+M1	Odyssey is useful and the next step is to perform an experimental study in real projects (i.e., industry).
8	Qi, i=4 to 9	M5+M4+M3 >= M2+M1	Odyssey is useful, but the next step is to analyze the participants' comments to identify and implement improvements before conducting an experimental study in real projects (i.e., industry).
9	Qi, i=4 to 9	M5+M4+M3 < M2+M1	Odyssey did not show evidence of usefulness. The next step is to review the tool features, as well as analyze the participants feedback to identify and implement improvements before conducting an experimental study in real projects (i.e., industry).
10	Qi, i=10 to 15	M5+M4 >= M3+M2+M1	Odyssey is easy to use and the next step is to perform an experimental study in real projects (i.e., industry).
11	Qi, i=10 to 15	M5+M4+M3 >= M2+M1	Odyssey is easy to use, but the next step is to analyze the participants' comments to identify and implement improvements before conducting an experimental study in real projects (i.e., industry).
12	Qi, i=10 to 15	M5+M4+M3 < M2+M1	Odyssey did not show evidence of ease of use. The next step is to search for usability standards, as well as analyze the participants' feedback to identify and implement improvements before conducting an experimental study in real projects (i.e., industry).

Expectations. It is expected that the collected data present insights that the training material was sufficient to conduct the evaluation, as well as that the similarity search mechanism meets the concepts of full and partial project-specific software process reuse. Finally, it is expected that the collected data present insights that Odyssey indeed helps the software process definition using SPrL.

Research Plan Analysis. In order to check the research plan and its artifacts, an evaluation step was applied. This step occurred iteratively involving three researchers. In this scenario, improvements have been made to mitigate the threats to validity, as well as reduce the time and effort required to accomplish the predefined tasks.

A lab package is available at https://doi.org/10.6084/m9.figshare.11893869. This package contains the participant characterization questionnaire, tasks document (two versions), evaluation questionnaire and the participants' analysis (full version).

4.2 Execution

Pilot. In order to analyze the plan feasibility, a pilot execution was conducted in early November 2018. It was performed by one participant (i.e., software process expert), a D.Sc. member of the Software Reuse Research Group (COPPE/UFRJ). The collected data of this pilot were not considered in the final data analysis.

Based on this pilot execution, some improvements were made: a) high-level tasks were divided into subtasks; b) complex tasks were simplified to specific tasks; and c) one observed defect was solved to avoid a possible impact.

Observation Study. After the pilot execution, the evaluation was conducted with seven software process experts (i.e., with knowledge in software process, as well as SPrL concepts) in December 2018, as described below.

Table 5 partially presents the participants' characterization (full characterization available in the lab package). These data were collected through the participant characterization questionnaire. The participants' experience in related activities was also collected.

Table 5. Participants' characterization.

Item	P1	P2	P3	P4	P5	P6	P7
Group Relationship	External	External	External	Former Member	External	Former Member	External
Academic degree	M.Sc.	M.Sc.	D.Sc.	PostDoc	PostDoc	D.Sc.	D.Sc.
Occupation	D.Sc. Student	Professor	PostDoc Student	Professor	Professor	Systems Analyst	Professor

The evaluation was performed based on steps specified in the planning phase, using communication templates (i.e., email), as well as predefined documents.

Initially, software process experts were mapped by convenience; fifteen experts were invited; seven experts scheduled a date to perform the evaluation; before the scheduled date, the experts received a training material; on the scheduled date, the expert received a consent form, participant characterization questionnaire and a training video focused on the supporting tool; after filling out these documents and watching the training video, the expert received the tasks document and remote access information; finally, at the end of the tasks, the expert received an evaluation questionnaire, which was sent to the researchers to finish the study.

4.3 Data Analysis and Results

Based on the interpretation model, the collected data were interpreted through the expressions aiming to analyze the results. Each specific goal is discussed below.

Goal Analysis (G1). This goal aims to evaluate the training material appropriateness.

Based on the collected data, two participants partially agree, four strongly agree and only one partially disagree. This can be interpreted as described in item 1 of Table 4 (M5 + M4 + M3 >= M2 + M1), indicating a positive result for the provided training material, where it has been considered sufficient to conduct the evaluation.

In this scenario, participant P6 highlighted concern about the participants' profile, selecting “partially disagree”. In the participant's opinion, the material may not be sufficient for those who do not have experience in software process variability. However, only software process experts were invited, all with experience in software process variability. In addition, despite having selected “strongly agree”, participant P4 indicated that the material could be simplified by reducing the training video duration.

Goal Analysis (G2). This goal aims to evaluate the similarity search mechanism support to the concept of full project-specific software process reuse.

Based on the collected data, four participants partially agree and three strongly agree. This can be interpreted as described in item 3 of Table 4 (M5 + M4 >= M3 + M2 + M1), indicating a positive result for the similarity research mechanism support, where it has been considered sufficient to meet the concept of full reuse.

In this scenario, according to participant P1 the similarity search mechanism support is relevant. However, the participant expressed concern about the use of the mechanism in real scenario (i.e., industry), with multiple disciplines and large amount of process elements.

In addition, participant P5 suggested allowing the user to select the context information that would be considered in the similarity calculation (i.e., subset). P6 participant recommended adding an explanation about how the similarity calculation works. Finally, participants P4 and P7 pointed out that the software process analysis and context comparison functionalities are very useful and informative.

Goal Analysis (G3). This goal aims to evaluate the similarity search mechanism support to the concept of partial project-specific software process reuse.

Based on the collected data, three participants partially agree, three strongly agree and only one partially disagree. This can be interpreted as described in item 5 of Table 4 (M5 + M4 >= M3 + M2 + M1), indicating a positive result for the similarity research mechanism support, where it has been considered sufficient to meet the concept of partial reuse.

In this scenario, although participant P3 selected “partially disagree”, no comments were provided in the questionnaire. The participant also showed no interest in analyzing decisions made in processes defined in a similar context, choosing to make his own decisions.

In addition, participants P1 and P7 expressed concern about the use of this mechanism in a real scenario (i.e., industry), with multiple disciplines and a large amount of process elements. Participant P5 suggested allowing the comparison of multiple cases (i.e., software processes) through a facilitating interface.

Finally, participant P4 highlighted that it is very positive to observe the choices (i.e., decisions made) in processes defined in a similar context, indicating that this kind of functionality can make the project manager more confident about his/her decisions.

Goal Analysis (G4). This goal aims to evaluate the perceived usefulness of the Odyssey tool. Fig. 3 presents the collected data focusing on the related questions (Q4-Q9) of this goal.

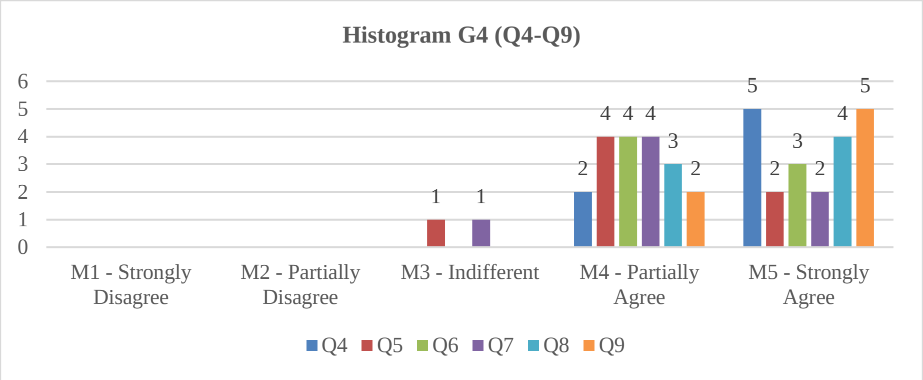

Fig. 3. Participants number X Selected option (G4).

Based on the data observed in Fig. 3, G4 goal can be interpreted as described in item 7 of Table 4 (M5 + M4 >= M3 + M2 + M1), where 21 + 19 >= 2 + 0 + 0 is true. Thus, these data present insights that: “Odyssey is useful and the next step is to perform an experimental study in real projects (i.e., industry)”.

In this scenario, participant P4 highlighted the usefulness of comparing context information in the similarity search mechanism. Finally, participant P6 pointed out that consulting similar cases is indeed useful.

Goal Analysis (G5). This goal aims to evaluate the perceived ease of use of the Odyssey tool. Fig. 4 presents the collected data focusing on the related questions (Q10-Q15) of this goal.

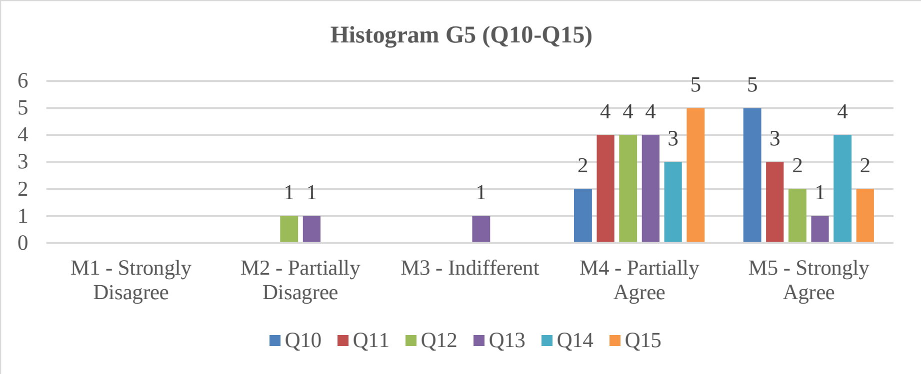

Fig. 4. Participants number X Selected option (G5).

Based on the data observed in Fig. 4, G5 goal can be interpreted as described in item 10 of Table 4 (M5 + M4 >= M3 + M2 + M1), where 17 + 22 >= 1 + 2 + 0 is true. Thus, these data present insights that: "Odyssey is easy to use and the next step is to perform an experimental study in real projects (i.e., industry)".

In this scenario, participant P1 pointed out that Odyssey main menu could be more flexible. Participant P5 suggested adding tutorials to guide the user during tasks. Finally, participant P7 emphasized the simplicity and objectivity of the tool, indicating that an industry study could help identifying tool improvements.

Improvements Identification. Although the collected data present insights of Odyssey's acceptance, an analysis was performed to find improvement opportunities. Thus, ten improvements were identified, which are presented in Table 6.

Table 6. Identified improvements.

Item	Description
1	Add custom menu on Odyssey main screen to offer specific options according to the selected item in the domain tree.
2	Customize the slicing mechanism to make possible to display of process elements specific information from the feature model.
3	Add confirmation message when populating context information if context information without assigned value exists.
4	Keep in memory the information context values in similarity search mechanism when used through other mechanisms.
5	Uncouple the similarity search mechanism when used through other mechanisms, in order to enable similar case analysis while doing a parallel task (e.g., select process elements).
6	Add explanation of how similarity calculation is performed in the similarity search mechanism.
7	Add an option to customize the weights of each similarity search mechanism's context information, as well as filter information by selecting a relevant subset for analysis.
8	Add complementary resources to assist in the visualization of similar case decision-making (e.g., simultaneous analysis of multiple cases).
9	Add tutorials to teach specific activity steps (e.g., software process definition).
10	Evaluate the adopted context information to better understand their different weights to support decision-making in the software process definition activity.

4.4 Threats to Validity

This evaluation was performed to characterize Odyssey acceptance. However, it is common that there are issues that may affect or limit the validity of the observed results. These issues are known as threats to validity [27]. Thus, in this section the threats to validity related to this work are presented.

- **Internal Validity:** The research plan proposes the use of software process experts, assuming that they represent a valuable profile. Besides that, the selection by convenience of the participants represents a threat, but to mitigate this threat, experts from different Brazilian states were invited to participate.
- **External Validity:** The low number, as well as the academic bias of the participants, limit the generalization power of the observed results. In addition, the predefined tasks focused on software process definition using SPrL delimit the study observations to this context (i.e., PSPE stage). In this scenario, only two mechanisms (i.e., slicing and similarity search) were used (Section 3). This delimitation was necessary to make the execution feasible. Besides that, the PSPE support represents the latest contributions to the Odyssey tool.
- ***Constructo* Validity:** The artifacts specified by the researchers (e.g., feature model, context characterization) represents another threat. Besides that, the use of the TeamViewer software to enable experts from different Brazilian states to participate through remote access also represents a threat, because of the risk of impact on the experts' perception.
- **Conclusion Validity:** The data analysis step did not use statistical methods, mainly due to the low number of participants, as well as the use of the Likert scale. In this scenario, to mitigate this threat, an interpretation model was specified in the planning phase to support the data analysis.

5 Conclusions and Future Works

This paper briefly describes a tool, called Odyssey, a software reuse environment. Besides that, an evaluation is presented, focused on characterizing Odyssey acceptance through its usefulness and ease of use by seven software process experts' analyses. In this scenario, the shared lab package can contribute to support future investigations of SPrL tools.

The collected data present insights of Odyssey's acceptance, meeting the study expectations. However, it is not possible to generalize these results based on the identified threats (Section 4.4).

A set of improvements was identified to enhance the support to SPrL engineering concepts based on participants' comments, as well as researchers' notes (Section 4.3) and are being implemented.

As future work, a study will be conducted in industry with the objective of verifying the benefits of Odyssey in real scenario, during SPDE and PSPE's activities, with participants of varied profiles and academic backgrounds. In addition, to increase Odyssey support to PSPE activities, other software process definition mechanisms will be proposed. Finally, SPrL evolution challenges can be investigated to propose a set of practices to assist SPrL artifacts modification.

Although the Odyssey tool is available as an academic project on the Software Reuse Research Group website[1], a release with the SPrL engineering support is still being prepared and is expected to be available by the end of the first semester of 2020.

Acknowledgment. The authors would like to thank CAPES, FAPERJ and CNPq for their financial support.

References

1. ALEIXO, F. A., FREIRE, M. A. SANTOS, W. C., et al., "A Model-driven Approach to Managing and Customizing Software Process Variabilities". In: *Proceedings of the 12th International Conference on Enterprise Information Systems* (ICEIS), v. 3, pp. 92-100, Madeira, Portugal, 2010.
2. BARRETO, A., DUARTE, E., ROCHA, A. R., et al., "Supporting the definition of software processes at consulting organizations via software process lines". In: *Proceedings of the Seventh International Conference on the Quality of Information and Communications Technology* (QUATIC), pp. 15-24, Porto, Portugal, 2010.
3. BARRETO, A., MURTA, L. G. P., ROCHA, A. R. C., 2011, "Software process definition: a reuse-based approach", *Journal of Universal Computer Science*, v. 17, n. 13, pp. 1765–1799.
4. BASILI, V., GREEN, S., LAITENBERGER, O., et al. "Packaging researcher experience to assist replication of experiments". In: *Proceedings of the ISERN Meeting*, pp. 3-6, Sydney, Australia, 1996.
5. COSTA, D. M., TEIXEIRA, E. N., WERNER, C. M. L., "Software Process Definition using Process Lines: A Systematic Literature Review". In: *Proceedings of the XLIV Latin American Computing Conference* (CLEI), São Paulo, Brazil. 2018.
6. COSTA, D. M., TEIXEIRA, E. N., WERNER, C. M. L., "Odyssey-ProcessCase: A Case-Based Software Process Line Approach". In: *Proceedings of the 17th Brazilian Symposium on Software Quality* (SBQS), Curitiba, Brazil. 2018.
7. DAVIS, F., 1985, *A technology acceptance model for empirically testing new end-user information systems: theory and results*. Doctoral Thesis, MIT Sloan School of Management, Cambridge, MA. USA.
8. DAVIS, F., 1989, "Perceived usefulness, perceived ease of use, and user acceptance of information technology", *MIS Quarterly*, v. 13, n. 3, pp. 319-340.
9. FISHBEIN, M., AJZEN, I., *Belief, attitude, intention and behavior: An introduction to theory and research*. 1 ed. California, Addison-Wesley. 1975.
10. FUGGETTA, A. "Software process: a roadmap". In: *Proceedings of the Conference on The Future of Software Engineering*, pp. 25-34, Limerick, Ireland, 2000.
11. HERNANDES, E., ZAMBONI, A., FABBRI, S., et al., 2012, "Using GQM and TAM to evaluate StArt - a tool that supports Systematic Review". *CLEI Electronic Journal*, v. 15, n. 1, pp. 13-25.

[1] http://reuse.cos.ufrj.br/

12. JAUFMAN, I., MÜNCH, J. “Acquisition of a Project-Specific Process”. In: *Proceedings of the International Conference on Product Focused Software Process Improvement* (PROFES), pp 328-342, Oulu, Finland, 2005.
13. JURISTO, N., MORENO, A. M., 2001, *Basics of Software Engineering Experimentation*. 1 ed. Springer.
14. LORENZ, W. G., BRASIL, M. B., FONTOURA, L. M., et al., 2014, "Activity-based software process lines tailoring". *International Journal of Software Engineering and Knowledge Engineering* (SEKE), v. 24, n. 09, pp. 1357-1381.
15. MAGDALENO, A. M., BARROS, M. O., WERNER, C. M. L., et al., 2015, "Collaboration optimization in software process composition". *Journal of Systems and Software*, v. 103, pp. 452-466.
16. MCIVER, J., CARMINES, E. G., *Unidimensional scaling*. London, UK, SAGE. 1981.
17. MURTA, L., BLOIS, A., LOPES, M., et al. “Run-Time Variability through Component Dynamic Loading”. In: *Tools Notes of the 18th Brazilian Symposium on Software Engineering* (SBES), pp. 67-72. 2004.
18. ODYSSEY, 2019, "Project Odyssey", In: *http://reuse.cos.ufrj.br/projetos/*.
19. OSTERWEIL, L. J., "Software processes are software too". In: *Proceedings of the 9th International Conference on Software Engineering* (ICSE), pp. 2–13, 1987.
20. ROMBACH, D. “Integrated Software Process and Product Lines”. In: *Unifying the Software Process Spectrum*, v. 3840, pp. 83-90, 2005.
21. SHULL, F., CARVER, J., TRAVASSOS, G. H., 2001, “An Empirical Methodology for Introducing Software Processes”, *ACM SIGSOFT Software Engineering Notes*, v. 26, n. 5, pp. 288-296.
22. SIMMONDS, J., PEROVICH, D., BASTARRICA, M. C., et al., "A megamodel for Software Process Line modeling and evolution". In: *Proceedings of the 18th International Conference on Model Driven Engineering Languages and Systems* (MoDELS), pp. 406-415, Ottawa, Canada, 2015.
23. SOLIGEN, R., BERGHOUT, E. *The Goal/Question/Metric Method: a practical guide for quality improvement of software development*. 1 ed. London, UK, McGraw-Hill Education, 1998.
24. TEIXEIRA, E. N., VASCONCELOS, A., & WERNER, C. M. L., “OdysseyProcessReuse - A Component-based Software Process Line Approach”. In: *Proceedings of the 20th International Conference on Enterprise Information Systems* (ICEIS), v. 2, pp. 231-238, 2018.
25. TEIXEIRA, E. N., ALEIXO, F. A., SOUSA, F. D., et al., 2019, “Software process line as an approach to support software process reuse: A systematic literature review”. *Information and Software Technology*, v. 116.
26. WASHIZAKI, H., 2006, “Building software process line architectures from bottom up”. *Product Focused Software Process Improvement*, v. 4034, pp. 415-421.
27. WOHLIN, C., RUNESON, P., HÖST, M., et al. *Experimentation in Software Engineering: An Introduction*. 1 ed. Springer. 2000.
28. XU, P., RAMESH, B., 2008, "Using Process Tailoring to Manage Software Development Challenges", *IT Professional*, v. 10, n. 4, pp. 39-45.