 doi:10.20944/preprints201809.0073.v1

Software Processes Analysis with Provenance

Gabriella C. B. Costa[1,2], Humberto L. O. Dalpra[1], Eldânae N. Teixeira[1], Cláudia M. L. Werner[1], Regina M. M. Braga[3], Marcos A. Miguel[4]

[1] COPPE - Federal University of Rio de Janeiro - RJ – Brazil
[2]Computing and Mechanics Department - Federal Center for Technological Education of Minas Gerais - MG - Brazil
[3] Computer Science Department - Federal University of Juiz de Fora – MG – Brazil
[4] Projetus Information Technology - P.O. Box 36120000 - Brazil
{gabriellacbc,humbertodalpra,danny,werner}@cos.ufrj.br,
regina.braga@ufjf.edu.br, marcos@projetusti.com.br

Abstract. Companies have been increasing the amount of data that they collect from their systems and processes, considering the decrease in the cost of memory and storage technologies in recent years. The emergence of technologies such as Big Data, Cloud Computing, E-Science, and the growing complexity of information systems made evident that traceability and provenance are promising approaches. Provenance has been successfully used in complex domains, like health sciences, chemical industries, and scientific computing, considering that these areas require a comprehensive semantic traceability mechanism. Based on these, we investigate the use of provenance in the context of Software Process (SP) and introduce a novel approach based on provenance concepts to model and represent SP data. It addresses SP provenance data capturing, storing, new information inferencing and visualization. The main contribution of our approach is PROV-SwProcess, a provenance model to deal with the specificities of SP and its ability in supporting process managers to deal with vast amounts of execution data during the process analysis and data-driven decision-making. A set of analysis possibilities were derived from this model, using SP goals and questions. A case study was conducted in collaboration with a software development company to instantiate the PROV-SwProcess model (using the proposed approach) with real-word process data. This study showed that 87.5% of the analysis possibilities using real data was correct and can assist in decision-making, while 62.5% of them are not possible to be performed by the process manager using his currently dashboard or process management tool.

Keywords: Software Process Analysis, Software Process Improvement, Data Provenance.

1 Introduction

During the software process (SP) , many different types of data can be generated and collected [11], such as: (i) Product Data: source code, configuration management data, documentation, executable codes, test suites, testing results, and simulations; (ii)

Process Data: explicit definition of a software process model, process enactment state information, data for process analysis and evolution, history data, project management data; and (iii) Organizational Data: ownership information for various project components, roles and responsibilities, and resource management data. Then, it is not a novelty that software development companies started to adopt data-driven practices in parts of their business over time [4]. However, the use of SP data remains a challenging topic for software engineers. Considering that engineering education "tends to focus on formulas, clear cause effect relations and predictable behaviors of the systems built by engineers, the notion of statistical behavior, analysis of large data sets and the use of averages and deviations feels less tangible, or, if nothing else, requires an alternative mindset from the people working with the data" [4]. Besides that, over time, the records accumulate, and the volume of data makes it difficult to conduct SP data analysis.

One possible way to support the analysis and verify the quality of SP generated data is by using provenance techniques and models. Data provenance can be defined as the description of the origins of a piece of data and the process by which it arrived in a database. It brings transparency and helps to audit and interpret data. Provenance has been successfully used in complex domains, like health sciences, chemical industries, and scientific computing [14]. The emergence of technologies such as Big Data, Cloud Computing, E-Science, and the increasing complexity of information systems further emphasize that traceability and provenance can be promising approaches.

Based on these facts and considering that SP is also a complex domain, the goal of this paper is to improve the process manager's understanding about the SP execution, providing analysis and decision-making possibilities using process data and provenance concepts. Then, our main research question is: *How can the use of provenance models and techniques in the SP domain support process managers analysis and data-driven decision making*? Then, we investigate the usage of provenance in the context of SP and propose a novel approach with a provenance model (called PROV-SwProcess) to deal with the specificities of SP. This approach addresses SP provenance data capturing, storing, new information inferencing and visualization. A difference of the proposed approach is its ability to infer new information, since it is ontology-based and uses an inference machine. In order to support process managers analysis and data-driven decision making, a set of SP analysis possibilities (e.g. process structure identification, possibilities for its redesign, understand stakeholder's involvement in process execution) were derived from PROV-SwProcess model and some insights of how to use them in decision-making are detailed. The current version of PROV-SwProcess model presented in this paper was carefully evaluated by three experts in process and provenance. Moreover, a case study was conducted in collaboration with a development company to instantiate PROV-SwProcess model (using the proposed approach) with real-word process data and the SP analysis possibilities were discussed.

The research methodology was undertaken in four steps (1) Research problem definition and a *quasi*-systematic review analyzing the use of provenance in SP. (2) The approach was specified and some studies to evaluate its viability were performed ([6][8]). (3) The core of the approach, PROV-SwProcess model, was defined and an evaluated by three experts in provenance and SP. (4) The approach was implemented with its tool support and a case study was performed.

This paper is organized as follows: A brief background considering SP and provenance is presented in Section 2, and Section 3 describes some related works. PROV-SwProcess model is presented in Section 4, with a discussion about the analysis possibilities derived from it. The approach that supports the model instantiation, new information inferencing and data visualization is presented in Section 5. Section 6 describes the model evaluation and a regular case study with a real-world process. Section 7 presents the paper conclusions.

2 Background

A well-defined SP should indicate the activities to be executed, the required resources, produced and consumed artifacts, adopted procedures (methods, techniques, models of documents, etc.), and the criteria for carrying out the activities [2]. The essential elements of SP considered in this approach are [12]: (i) **Activity**: deals with the process activities used to create and/or maintain software and how they compose the SP; (ii) **Stakeholder**: refers to organizations, persons, projects, or teams acting or interesting in the software process activities; (iii) **Resource**: involves hardware equipment and software products used by the activities; (iv) **Procedure**: relates to methods, techniques and document templates adopted by the software process activities; and (v) **Artifact**: represents different types of objects produced, changed, and used in process activities.

During the process execution, SP data are captured and analyzed during the process evaluation. Process analysis (or evaluation) can be of two different types [26]: (i) Deductive Analysis: considers an abstract specification of a process in some formal logic, aiming to discover inconsistencies or anomalies that would be present in enactments of the process; or (ii) Retrospective Analysis: analyze empirically gathered data from several enactments of a process, to discover patterns of anomalous behavior. Our approach focuses on retrospective analysis, i.e., on SP execution data.

Data provenance can be defined as the origins description of a piece of data and their processing history [14]. Provenance differs from traditional data items and meta-data considering that it is an immutable directed graph, incrementally captured at run-time [23]. Nevertheless, process data provenance capturing does not interfere in the SP execution and allows the process managers or process data analysts to refine the applied filtering rules for data process collection [15].

According to Freire *et al.* [14], when we have provenance from computational tasks, it can be divided into two types: (i) prospective provenance, that captures a computational task's specification and corresponds to the steps that must be followed to generate a data product, and (ii) retrospective provenance, that captures the steps executed as well as information about the environment used to derive a specific data product.

To obtain the benefits of provenance information, data provenance should be captured/stored in an integrated manner to allow queries on that data. In this vein, there are two main models proposed in the literature: Open Provenance Model (OPM) [20] and, more recently, W3C PROV model [18]. In this paper, PROV was chosen and extended, considering that it is a standard model provided by W3C and it has causal relationships that are not explicit in OPM. PROV model [18] aims to express data provenance

through the description of entities, activities, and agents involved in producing or delivering an object, and the causal relationships between them. The seven main PROV causal relationships are: (1) used, (2) wasGeneratedBy, (3) wasAssociatedWith, (4) wasAttributedTo, (5) actedOnBehalfOf, (6) wasDerivedFrom, and (7) wasInformedBy.

3 Related Work

Our approach differs from the other process analysis approaches based on process execution logs [1][2][5] since it addresses the possibility of deriving implicit knowledge, using an ontology, inference rules, and an inference machine. In this vein, causal relationships between the process execution data can be automatically inferred, even if it has not been provided (e.g., artifacts creation and modification by stakeholders, derivation between artifacts, usage of specific procedures to develop an artifact). Considering this fact, related work is analyzed through two different perspectives: a) provenance data models that are extensions from PROV (considering that PROV-SwProcess model[1] is an extension of PROV), and b) the use of provenance in the context of SP.

Considering that PROV model is generic and presents several possibilities of causal relationships, there are in the literature some proposals to adapt this model to specific domains, such as D-PROV [19] and ProvONE [7]. D-PROV extends PROV to represent the process structure, i.e., to enable prospective provenance storage and query. D-PROV was a previous incarnation of ProvONE, which is a model for scientific workflow provenance and extends PROV with its specific structure elements. Although these models are useful in scientific workflow domain and process in general, it does not suffice for capturing and analyzing provenance in the SP domain. For example, in ProvONE, the workflow execution corresponds to the execution of computational tasks only by software agents but, in the SP, we need to express different types of agents, such as, persons, teams, and organizations. Besides, ProvONE does not have specific types of procedures and artifacts and does not propose new rules to derive implicit provenance information. Considering the gaps of ProvONE and the fact that PROV does not capture the specificities of SP, extensions in this model should be made. An initial effort in this regard was made in previous works [6][8].

Considering the use of provenance in SP, it was found in a previous literature review, that the application and use of provenance data in the SP domain were mentioned for the first time in 2005 [27] and all others were published from 2007 onwards. One of the possibilities regarding a greater number of publications appearing after 2007 is due to the emergence of the Provenance Challenge, started in 2006 [21]. However, it should be considered that this event addressed the provenance challenges in the general scope and not specifically in the SP domain. The results dating from only 2005 also shows the lack of maturity of this research field and the need, as underscored by some authors [9][10][16], for more scientific papers about using provenance in the context of SP.

A code provenance management tool called Ariadne is proposed in [9]. It tracks the provenance of source code and generates provenance reports to facilitate the

[1] It is detailed in Subsection 4.1.

management of its intellectual property. Other works, such as [10][16], motivate the need to model and extract software artifacts provenance. Davies et al. [10] explore the recovery of the provenance of software artifacts by a broad set of techniques (signature matching, source code fact extraction, software clone detection, call flow graph matching, string matching, historical analyses etc.) and Godfrey [16] cites the PROV model specification and shows a motivating example that uses hashing to quickly and accurately identify version information of embedded Java libraries. Although these works deal with provenance in the context of software development, they do not address the provenance of SP as a whole. They focus on software artifacts or source code. In PROV-SwProcess, we treat not only the artifacts, but the activities, agents and the various relationships that can be established in SP. A technique called PRiME [17] also adapts projects to interact with a provenance layer. Based on PRiME, Wendel et al. [25] present a solution to failures in software development processes, using the Open Provenance Model and SOA architecture. However, the last two works do not specify how data provenance can be inferred and used to support SP analysis and data-driven decision-making as done in our approach. The most recent publication in this scenario [13] starts a discussion of using complex networks concepts (besides an ontology) to help in SP data interpretation aiming to support in SP improvement. However, it does not address specific concepts of SP as we have done in PROV-SwProcess Model (it uses ProvONE) and does not provide the analysis discussed in our work.

4 Provenance in Software Processes

Based in a previous literature review (whose main points were presented in Section 3), there is no consensus regarding the most appropriate provenance model to be used specifically in SP domain. The model most used in the provenance area is PROV. However, the direct application of this model to SP domain lacks in capturing some SP specificities such as Resources and Procedures used or adopted by the activities, different types of SP artifacts (e.g., software product, software items and models), as well as new possible relationships between them. To overcome this gap and considering the existence of different systems that can be used during SP execution (e.g., version control system, issue trackers, and documentation management systems) without a standard model to capture the provenance of these processes execution, PROV-SwProcess model was defined and described in the next subsection.

4.1 PROV-SwProcess: A PROV Extension Data Model for Software Processes

PROV-SwProcess model was developed to be a standard for SP provenance representation. It was defined as a PROV extension, aiming to capture and infer relevant information about SP data.

A preliminary proposal of PROV-SwProcess (called PROV-Process) was published in 2016 [8]. It is an initial approach to apply the PROV model in SP domain. PROV-SwProcess aims to incorporate the basic ideas of this work, as well as additional contributions, to derive an adequate standard that can be used in SP.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 September 2018 doi:10.20944/preprints201809.0073.v1

PROV-SwProcess covers prospective and retrospective provenance [14] and the essential aspects of SP: activities, stakeholder, resource, procedure, and artifact [12]. It is divided into (i) associations (or relations), (ii) classes, and (iii) specific inference rules. Figure 1 describes PROV-SwProcess[2], focusing its Retrospective Provenance[3] part and using a diagram to represent its conceptual model. The following points should be considered regarding it: (1) Constructs and associations presented between "<<>>" were derived from PROV. For example: the <<Activity>> class corresponds to the Activity PROV type. Newly PROV-SwProcess associations/relations and classes appeared without "<<>>"; (2) Elements in ellipses are specializations of the Entity PROV type and elements in pentagons are specializations of the Agent PROV type; (3) Associations with solid lines are used to capture Retrospective Provenance and associations with dashed lines can be inferred by PROV-SwProcess approach and their respective provenance rules, that is, they do not necessarily need to be captured or informed in the SP provenance data. The data are transformed into an ontology that enables to make inferences into the data using a reasoner; (4) All PROV-SwProcess relations have a related inverse relation (for example: the inverse relation of <<Used>> is the relation <<WasUsedBy>>), however, these were not explicit in the figures aiming to facilitate the understanding of the proposed model; (5) When there is more than one SP instance to be analyzed, the relation WasComposedBy can also be inferred, allowing to obtain all the stakeholders, resources, artifacts, and procedures involved in a SP instance.

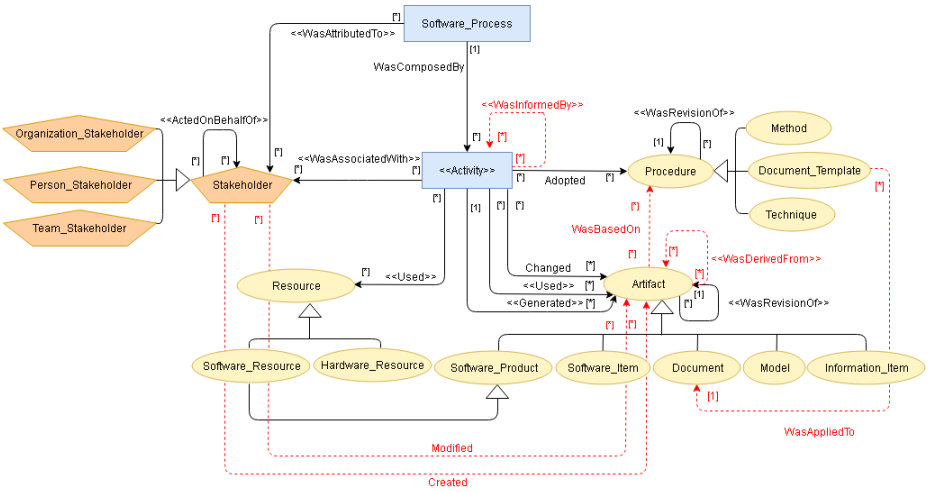

Fig. 1. PROV-SwProcess Model (Retrospective Part)

PROV-SwProcess Ontology and Inference Rules

PROV-SwProcess model has also an ontology that extends PROV-O ontology [18] and is specified using Ontology Web Language (OWL2)[4]. We adopted an ontology to

[2] The complete model specification can be accessed at: http://bit.ly/provswprocess

[3] PROV-SwProcess defines both Retrospective and Prospective Provenance, however, due to space restrictions, we focused on its Retrospective part in this paper.

[4] http://bit.ly/provswprocessontology

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 September 2018 doi:10.20944/preprints201809.0073.v1

support our model and approach, considering that it addresses the possibility of deriving implicit knowledge, using some inference rules and an inference machine or reasoner (as an example, in Fig. 1, all the associations with red dashed lines are inferred even it was not provided in the process data execution).

An inference rule can be applied to PROV-SwProcess instances to add new PROV-SwProcess statements, bringing implicit information. The inferences rules have been defined and specified using the Semantic Web Rule Language (SWRL), specifically to the SP domain. They can be divided into 8 groups: (1) *Created*, (2) *Modified*, (3) *WasBasedOn*, (4) *WasAppliedTo*, (5) *WasDerivedFrom*, (6) *WasInformedBy*, (7) *WasComposedBy*, and (8) *HadRole*. All the proposed inferences have the form:

IF A1 and ... and Ap THEN there exists y1...ym such that B1 and ... and Bk

That means: ∀ x1,....,xn. A1 ∧ ... ∧ Ap ⇒ ∃ y1...ym . B1 ∧ ... ∧ Bk, where x1...xn are the free variables of the inference

As an example, an inference rule of the *Created* group is[5]:

IF wasAssociatedWith(_ass; ac,sta,_attrs1) and generated(_gen; ac,art,_attrs2) THEN there exists _id such that created(_id; sta,art,[]).

This inference states that if an activity *ac* was associated with a stakeholder *sta* and this activity *ac* generated an artifact *art*, the relation *created* between the stakeholder *sta* and the artifact *art* can be inferred. Figure 2 shows an example to explain PROV-SwProcess model possible inferences (the inferred associations appear in red). Even if there is no explicit and direct relation in the provenance data between Mary and Payment_Test_Cases, we can infer, using the rule presented, that *Mary* created *Payment_Test_Cases*.

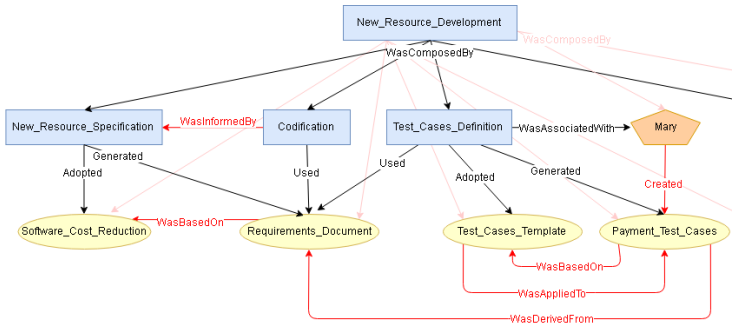

Fig. 2. PROV-SwProcess Inferences Example

In order to achieve the main objective of the approach (improve the process managers' understanding about the SP execution, providing analysis and decision-making possibilities using process data and provenance concepts), some specific goals were derived from PROV-SwProcess model and are described in the following.

[5] All the inferences rules are detailed in the complete PROV-SwProcess model specification.

4.2 Software Process Analysis Goals

Aiming to support process managers' analysis and data-driven decision making, a set of SP analysis possibilities, divided into specific goals, was derived from PROV-SwProcess model and some insights of how to use it in decision-making are detailed. Due to space restrictions, these analyses represent an initial set of how to apply the resources provided by the approach, and do not cover the whole model.

- **Goal 1: Process Structure Identification and possibilities for process redesign**
  - **Question 1.1**: What are the process activities, artifacts, resources, procedures, stakeholders, and the relations among them?
    - **How to answer the question?** Using a list or a graph with the executed activities, artifacts, resources, procedures, and stakeholders with its relations.
    - **Analysis**: It is possible to identify all the process elements that participated in process executions (or in some process instance) and the relations among them.
    - **Decision-Making Possibility**: After identifying the process elements and the relations between them it is possible to find gaps (elements without association or inadequate relation stablished) in the analyzed data and try to correct it in next process executions.
  - **Question 1.2:** Which procedures are used by the process?
    - **How to answer the question:** Using the number of procedures used to the develop the process artifacts and a list or graph with them.
    - **Analysis**: It is possible to check which procedures influenced an artifact development; Verify the procedures most useful in the analyzed instance(s), when a procedure is used by artifacts in a number greater than the average; Check procedures useless, i.e., although existing, these procedures were never used during the execution of the processes carried out by the organization.
    - **Decision-Making Possibility**: When verifying that procedures influenced an artifact development, the process manager can evaluate if this fact was really planned/expected (in process modeling phase) or not; if this information is not specified in the process model, the process manager may include it; Being aware that a procedure is widely used by the process instances, the manager can better plan any changes in this procedure, since this can have a great impact on future executions; If a procedure has not been used during process execution, this in-formation may be valid for the process manager to evaluate whether this procedure needs to be changed/reshaped to be used as planned or if it should be removed from the process. Another point of analysis would be the impact of not having a standard for the development of some artifacts – it could impact the quality level of generated artifacts, as well as cause errors by the difficulty of understanding some information in these artifacts, etc.
  - **Question 1.3:** Which activities has a high complexity?
    - **How to answer the question:** Using the number of Artifacts, Stakeholders, Procedure and Resource associated to a specific activity or a graph showing these relations.
    - **Analysis**: It is possible to check when activities are associated with many stakeholders, artifacts, procedures, and resources, when compared to the other activities, indicating that an activity could be more complex than others.

 doi:10.20944/preprints201809.0073.v1

- **Decision-Making Possibility**: With the information provided by the analysis presented above, the process manager can evaluate if this fact was really planned/expected (in process modeling phase) or not; if this information is not specified in the process model, the process manager may change the process model to better represent the process that was in fact executed; A possible evaluation of the activities detected as more complex can be performed, aiming to divide it into less complex sub activities.

— **Question 1.4:** Which activities has a high dependency?
- **How to answer the question:** Using the number of dependent activities of each executed activity and a list of them or some graph representation showing these dependencies.
- **Analysis**: It is possible to analyze the dependency between two activities, i.e., when occurred the exchange of some artifact by two activities, one activity using some entity generated or changed by the other. It is also possible to discover which activity occurred before or after another during execution time and to identify possible bottlenecks based on activities dependency.
- **Decision-Making Possibility**: From the previous analyzes, the process manager can confront the activities (and its flow) specified in the process model and how they occurred during execution. If there are any discrepancies, he can make changes in the process model, according to what he verified that, in fact, it was executed. Another decision is trying to make changes in the process model to avoid bottlenecks, if it were identified in the previous analysis.

- **Goal 2: Understand stakeholder's involvement in process execution**

— **Question 2.1**: What is the activities distribution among stakeholders?
- **How to answer the question**: Number of activities each stakeholder is involved and a list or some graph representation with them.
- **Analysis**: It is possible to discover, from a stakeholder, all the activities (and the total of these activities) in which he/she participated, allowing to understand the activities distribution among stakeholders in the process execution.
- **Decision-Making Possibility**: When verifying that a stakeholder is participating in much activities than others, the process manager can evaluate if this fact was really planned/expected (considering, for example, that a stakeholder was associated to a high number of activities because him/her always is attributed to activities with a lower level of complexity) or if it has been occurring due to an inadequate activity distribution during the process instantiation.

— **Question 2.2**: Which artifacts are known by a stakeholder, considering that in some process execution he/she created or modified such artifact?
- **How to answer the question**: Number of artifacts each stakeholder is involved in its creation or modification and a list with them or some graph representation showing stakeholders x artifacts.
- **Analysis**: It is possible to discover all the artifacts that were created and/or modified by a stakeholder, allowing to understand about what artifacts this stakeholder has some knowledge, considering he/she manipulated this artifact in some process execution. Considering the artifact view point, it is possible to

discover all the stakeholders that has some knowledge about it, considering it was created or modified by them.
Decision-Making Possibility: in a future instantiation of the analyzed process, if a certain task is associated with a specific artifact, the process manager (or the responsible for the process instantiation) can allocate to this task a stakeholder with greater or less knowledge about the artifact to be manipulated during this task execution, according to the project objectives / goals.

- **Question 2.3**: What are the relationships among stakeholders?
  - **How to answer the question**: Number of responsibility relation among stakeholders and a list of them or some graph representation showing stakeholders responsibility relations.
  - **Analysis**: It is possible to know the responsibility between the stakeholders during a process instance execution, detecting whether one stakeholder is responsible for many others or not.
  - **Decision-Making Possibility**: after analyzing the responsibility among stakeholders in executed instances, the process manager can use this information when allocating the responsibilities between stakeholders when a new instance of this process is created, according to the project objectives / goals.
- **Question 2.4**: Which roles each stakeholder assumes?
  - **How to answer the question:** Number of roles performed by a stakeholder and a list of them or some graph representation showing stakeholders x roles.
  - **Analysis**: It is possible to analyze all the roles played by a specific stakeholder as well as, from a role, to verify which stakeholders can accomplish it.
  - **Decision-Making Possibility**: In a next instantiation of this process, if the process manager needs to allocate some person stakeholder in a specific activity that needs some pre-defined role, he can evaluate who can perform this role, based on stakeholders' skills. On the other hand, he can also decide who should participate in a training programming in order to be able to accomplish more roles during process execution.

Considering the presented model and its analysis possibilities, next section presents the approach that supports the model instantiation.

5 Approach

In our vision, the best way to capture the SP provenance data is adapting the process execution engine or the workflow engine used by the organization to collect provenance data (as it is done in cases of scientific workflows). However, most small and medium-sized companies, in the initial levels of software maturity models, do not use such tools to execute their software processes, but rather a set of different tools (e.g., version control system, issue trackers, and documentation management systems). Considering the diversity of such tools, a wrapper should be developed to structure all the recorded execution data according to PROV-SwProcess Model. This is the initial effort required to use our approach. Considering this fact, the approach that supports the model instantiation, new information inferencing and data visualization is composed by three main elements: (i) SP provenance data capture and storage; (ii) Deriving SP implicit

information using inference mechanisms; (iii) Converting SP provenance data into a graph format aiming to facilitate process manager in a decision-making activity. These three main elements use as basis PROV-SwProcess model presented in Subsection 4.1.

Approach execution has five activities: (1) Process execution and provenance data capture; (2) Data transformation according to the PROV-SwProcess model; (3) Data storage and ontology generation; (4) Inference machine execution; and (5) Data visualization and analysis. These activities must be carried out sequentially. Considering the first activity, a set of execution data is requested for each of the analyzed processes:

1. *Performed SP instance with its name and responsible (a Stakeholder);*
2. *Performed activities of the SP instance with its name, start, and end time;*
3. *Stakeholders associated with the performed activity and their respective roles;*
4. *Artifacts changed, used, or generated by the performed activity;*
5. *Procedures adopted for the execution of the performed activity (optional);*
6. *Hardware and/or Software resources used by the performed activity (optional);*
7. *Process model to capture prospective provenance (optional).*

Although data from items 5 and 7 are optional, it is important to note that to achieve a more accurate and specific data analysis, it is important to report as much data and information as possible. If the data captured in the first activity are not previously organized according to the PROV-SwProcess model, they must be manipulated and organized/stored according to this model. After storing the SP data, an ontology is generated with them and an inference machine is executed. Lastly, a graph visualization using all the data and new inferred information is generated to allow process manager analysis and support data-driven decision-making. A tool that supports the execution of the proposed approach was implemented as a web application.

Finally, we should point that some training about the visualization tool support is required, to show to the process manager how to use it to obtain the proposed analysis. SP should not be changed to use the approach and it could be used to any kind of software process.

6 Evaluation

Initially, an evaluation in a survey format was made with experts in provenance and software processes, to verify and correct PROV-SwProcess concepts, relations, and inferences possibilities (Subsection 6.1). After that, a case study was conducted in collaboration with a software development company to instantiate PROV-SwProcess model (using the proposed approach) with real-word process data (Subsection 6.2).

6.1 Evaluation with Experts

PROV-SwProcess model presented in this paper is in its third version. It was generated after two rounds of an evaluation with specialists in SP and data provenance.

In the first round, two experts in software process and data provenance with PhD degree evaluated the first version of PROV-SwProcess model. The evaluation was performed based on a questionnaire containing 32 Discrepant Cases (DCs) to be analyzed. DCs are issues suggesting defects or general situations in which defects can be detected

[22] and making explicit for the reviewers the perspectives to look for defects. The definition of DCs to compose the questionnaire intended to cover all the PROV-SwProcess elements follows a defect taxonomy [24]. A question example from the questionnaire is: "Is some association needed to describe a performed software process (in addition to *wasAttributedTo* and *wasComposedBy*) omitted from the model?". The specialists could answer *Yes*, *No*, or *I don't know / I am not sure*. *Yes* as an answer means that the expert has found some semantic defect in the model. In these cases, a justification was requested. Then, based on this explanation, some alteration in PROV-SwProcess was evaluated, trying to solve the defect. When the expert answers *No*, it means that the element in evaluation has no semantic defect. *I don't know / I am not sure* was applied when the expert had doubts about some specific element. After receiving the expects questionnaire, a direct conversation with the specialist was conducted to understand the expert reasoning and what could be done in the model to eliminate errors found and uncertainties. During this round, 'Participant 1' found 9 defects (out of 32 DCs) and presented 2 uncertainties, while 'Participant 2' found only 1 defect. Analyzing these numbers, it is possible to note that the percentage of defects found was much lower than the number of correct elements in the model (81% of correct items versus 16% of defects and 3% of uncertainties), however, we considered the need for a re-evaluation of the model generated after this first evaluation round.

The second round follows the same format of the first, with a different expert (with PhD degree and good knowledge in provenance and SP). Some adjustments were made to the form to accommodate the model corrections, e.g., new added relations/concepts. This evaluation form has 38 questions and the expert pointed out 32 correct points and 6 defects (2 omissions and 6 incorrect facts). We corrected these points and generated the version presented in this paper. Although a new analysis of this third version was not performed by a fourth expert, we chose to evaluate this last version through an instantiation of the model with real data, as will be presented in next subsection.

6.2 Evaluation using real-world data

Considering the proposed approach, we are interested in evaluating its feasibility in real world contexts. In this vein, a case study was conducted in collaboration with a development company to instantiate PROV-SwProcess model (using the proposed approach) and check SP analysis goals (presented in Section 4.2) using real-word process data.

Study definition

The evaluation scope was defined based on GQM method [3]: **Analyze** the proposed approach and PROV-SwProcess provenance model to evaluate its feasibility **for the purpose of** supporting data analysis and data-driven decision making **with respect to** provide relevant information **under the point of view of** process managers **in the context of** software process. From the scope definition, the research question is as follows: *How can the use of provenance models and techniques in SP domain support process managers analysis and data-driven decision making?* Our study proposition is: *PROV-SwProcess model (and its tool support) can improve the process manager's understanding about the SP execution, providing analysis and decision-making possibilities*.

Planning

 doi:10.20944/preprints201809.0073.v1

Context Selection: The study was based on real process execution data, collected from a development process in a medium software development company. This company is specialist in developing accounting systems and solutions and has been acting in a national market for 25 years.

Data Collection: The data were collected using a direct observation. Researchers had a direct contact with the subject using a semi-structured interview and a questionnaire to check the results when using the approach with collected process execution data.

Instrumentation: The following instruments were selected: Consent form from the company and the subject, to allow the publication of the collected data in this work. Profile subject background questionnaire. Questionnaire used during the interview to evaluate the correctness / usefulness of the approach analysis possibilities.

Study Execution

Goals: This experiment aims to evaluate PROV-SwProcess model (and the proposed approach) in supporting SP analysis and decision-making using process execution data from ten random instances of a real-world SP.

Subject Characterization: The subject is a male, 40 years old, who works in company as a SP manager for ten years and has a broad knowledge of the analyzed process.

Scenario: The analyzed data is from a process that deals with error handling and the implementation of new features in an ERP Project. It is performed by six different roles (Client, Test Team, Support, Support Manager, Development Manager, and Programmer) and has five activities: *System Error Report, New Feature Request, Case Registration, Case Resolution,* and *Close the Case.*

Execution: The following steps were conducted: (1) process execution data extraction and structuring according to PROV-SwProcess model (a wrapper was developed for Mantis and a proprietary VCS); (2) data upload in the tool support; (3) using approach data visualization module to generate the visualizations that assist in the SP analysis, and (4) validation of the obtained analyzes with the process manager, through a semi structured interview and a questionnaire. As an example, considering SP Analysis Goal 2 (*Understand stakeholder's involvement in process execution*), the generated provenance graph to assist in answering question 2.4 *(Which roles each stakeholder assumes?*) is shown in Figure 3. Stakeholders are represented by the orange pentagons, activities are the blue rectangles, and the roles are the yellow ellipses. Using this figure, we can see all the stakeholders that acts as a *Programmer*, as *Support* or as a *Client* (their names were omitted for confidentiality reasons). The group of roles in the lower corner of the figure corresponds to three roles informed in the process model which had no associated stakeholder. According to this figure, we can see, for example, that the most versatile stakeholder is *Person_1*, that acts as *Programmer* and as *Support*. Considering the decision-making possibilities about this question, in a next instantiation of this process, if the process manager needs to allocate a *Programmer* or a *Support* person in a specific activity, he knows who can perform these roles. In addition, he can verify why there are three roles not performed during the analyzed instances. All the other questions (proposed in Section 4.2) were analyzed during the interview.

Results and discussion: All the SP analysis goals and questions were performed. Table 1 presents a summary of the obtained results during the interview (87.5% of the analysis possibilities was correct and can assist in decision-making, while 62.5% of them are not possible to be performed by the process manager using his currently dashboard or process management tool). Considering these results, we can see that only the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 4 September 2018 doi:10.20944/preprints201809.0073.v1

analysis of question 1.3 was not considered correct (the subject said that activities complexity is not easy to measure and other aspects should be considered, like activities time duration).When checking question 1.1, the subject said that without our approach he can obtain all the process elements, however, the relations among them are not explicit in his currently process dashboard as in our approach or he takes much time to obtain it using complex SQL queries. Considering question 2.2, he mentioned that he can do this analysis using some SQL query, however, he could not obtain a visualization that facilitates the analysis, as in our approach.

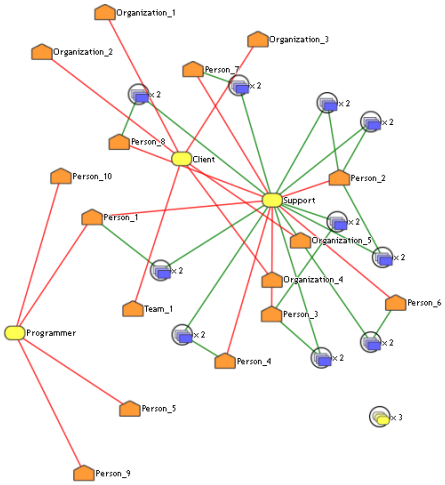

Fig. 3. Provenance Graph to Support Question 2.4.

Table 1. Evaluation of the Goals and Analysis Using SP Data

Goals	Questions	(1)	(2)	(3)
1	Question 1.1	Yes	Yes	Partially
	Question 1.2	Yes	Yes	No
	Question 1.3	Partially	No	No
	Question 1.4	Yes	Yes	No
2	Question 2.1	Yes	Yes	Yes
	Question 2.2	Yes	Yes	Partially
	Question 2.3	Yes	Yes	No
	Question 2.4	Yes	Yes	No

(1) Is the provided analysis correct? **(2)** Can the provided analysis assist in decision-making? **(3)** Does your current process management tool/dashboard provide this kind of analysis?

6.3 Threats to Validity

Considering the evaluation with experts, they were defined according to their knowledge in the approach related areas (SP and provenance) and not using a random selection. In addition, their evaluation was performed offline, without any follow-up from the researchers. Considering the case study, it can be considered as a first step of the approach evaluation in real scenarios, since the number of case study and subject

are not ideal, especially from a statistical point of view. Further study is already being conducted and could provide additional evidence that was not observed. Despite of that, additional evaluations are still necessary, considering other SP contexts and larger cases, aiming to extend the validity of the approach to SP in general. Additional aspects such as non-functional requirements, e.g., performance and scalability, were not considered in the presented study, however, they show preliminary evidence of the approach benefits in SP analysis and decision-making.

7 Conclusions

Considering the research question *How can the use of provenance models and techniques in the SP domain support process managers analysis and data-driven decision making?*, our main goal consists in providing an approach that *uses provenance models and techniques in SP domain to support process managers analysis and data-driven decision making*. PROV-SwProcess model and an approach to support its instantiation and process data analysis was presented and evaluated by experts and using a case study. An initial set of eight questions was defined based on process goals and some analysis and decision-making possibilities were discussed. While the expert's evaluation allowed corrections and improvement points on the provenance model, the case study showed that 7 out of 8 analysis using real data was correct and can assist in decision-making, and 5 of them are not possible to be performed by the process manager using his currently dashboard or process manager tool. Based on this study, we obtained preliminary evidences that *PROV-SwProcess model (and its tool support) can improve the process manager's understanding about the SP execution, providing analysis and decision-making possibilities*. Future researches can arise from this work. Initially, further studies should be performed to analyze the approach using other process/scenarios, as well as the definition and evaluation of SP analysis goals and questions using prospective provenance. Improvements in the visualization mechanism can be done aiming to consider other information, e.g., activities execution ordering and spent time.

References

[1] Aversano L et al (2004) Managing coordination and cooperation in distributed software processes: the GENESIS environment. Software Process: Improvement and Practice, 9, pp. 239-263.

[2] Avrilionis D et al (1996) A unified framework for software process enactment and improvement. Proceedings of Software Process. IEEE Comput. Soc. Press, pp. 102-111.

[3] Basili V et al (1994) Goal Question Metric Paradigm. Encyclopedia of Software Engineering, v. 1, edited by John J. Marciniak, John Wiley & Sons, pp. 528–532.

[4] Bosch J (2017) Speed, Data, and Ecosystems: Excelling in a Software-Driven World. CRC Press.

[5] Cook JE (2000) Software process analysis. ACM SIGSOFT Software Engineering Notes. 25, 1, pp. 44.

[6] Costa GCB et al (2016) Software Process Performance Improvement Using Data Provenance and Ontology. Lecture Notes in Business Information Processing. Springer International Publishing, pp. 55–71.

[7] Cuevas-Vicenttín V et al (2016). ProvONE: A PROV Extension Data Model for Scientific Workflow Provenance. Retrieved July 2018 from https://purl.dataone.org/provone-v1-dev
[8] Dalpra HL et al (2015) Using Ontology and Data Provenance to Improve Software Processes. In Brazilian Ontology Research Seminar, São Paulo, Brazil, pp. 10-21.
[9] Dang YB et al (2008) A code provenance management tool for ip-aware software development. Proceedings of the 13th International Conference on Software Engineering, ACM, pp. 975-976.
[10] Davies J et al (2012) Software Bertillonage. Empirical Software Engineering, 18, 6, pp. 1195–1237.
[11] Derniame JC et al (1999) Software Process: Principles, Methodology, and Technology. Springer Berlin Heidelberg.
[12] Falbo RDA, Bertollo G (2009) A software process ontology as a common vocabulary about software processes. International Journal of Business Process Integration and Management, 4, 4, pp. 239-250.
[13] Falci MF et al (2018) Software Process Improvement through the Combination of Data Provenance, Ontologies and Complex Networks. Proceedings of the 20th International Conference on Enterprise Information Systems, 2, pp. 61-70.
[14] Freire J et al (2008) Provenance for Computational Tasks: A Survey. Computing in Science & Engineering, 10, 3, pp. 11–21.
[15] Ghoshal D, Plale B (2013) Provenance from log files. Proceedings of the Joint EDBT/ICDT 2013 Workshops on. ACM, New York, NY, USA, pp. 290-297.
[16] Godfrey MW (2015) Understanding software artifact provenance. Science of Computer Programming, 97, pp. 86–90.
[17] Miles S et al (2011) PrIMe. ACM Transactions on Software Engineering and Methodology. 20, 3, pp. 1-42.
[18] Missier P et al (2013) The W3C PROV family of specifications for modelling provenance metadata. Proceedings of the 16th International Conference on Extending Database Technology, ACM, pp. 773-776.
[19] Missier P et al (2013) D-PROV: extending the PROV provenance model with workflow structure. In Proceedings of the 5th USENIX Workshop on the Theory and Practice of Provenance (TaPP '13). USENIX Association, Berkeley, CA, USA, pp. 9:1–9:7.
[20] Moreau L et al (2011) The Open Provenance Model core specification (v1.1). Future Generation Computer Systems, 27, 6, pp. 743-756.
[21] Moreau L et al (2008) Special Issue: The First Provenance Challenge. Concurrency and Computation: Practice and Experience, 20, 5, pp. 409-418.
[22] Shull F et al (2000). How perspective-based reading can improve requirements inspections. IEEE Computer, 33, 7, pp. 73-79.
[23] Sun L et al (2013) Engineering access control policies for provenance-aware systems. Proceedings of the third ACM conference on Data and application security and privacy. ACM, pp. 285-292.
[24] Teixeira EN et al (2015) Verification of Software Process Line Models: A Checklist-based Inspection Approach. Proceedings of XVIII Ibero-American Conference on Software Engineering, Peru, Lima, 2015.
[25] Wendel H et al (2010) Provenance of Software Development Processes. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 59-63.
[26] Wolf AL, Rosenblum DS (1993) A study in software process data capture and analysis. Proceedings of the Second International Conference on the Software Process-Continuous Software Process Improvement. IEEE, pp. 115-124.
[27] Xu P, Sengupta, A (2005). Provenance in Software Engineering - A Configuration Management View. Proceedings of the Eleventh Americas Conference on Information Systems (AMCIS), Omaha, NE, USA, pp. 3103-3107.