Saimir Bala

Mining Projects from Structured and Unstructured Data

Article (Draft)

Mining Projects from Structured and Unstructured Data

Saimir Bala[1]

Abstract: Companies working on safety-critical projects must adhere to strict rules imposed by the domain, especially when human safety is involved. These projects need to be compliant to standard norms and regulations. Thus, all the process steps must be clearly documented in order to be verifiable for compliance in a later stage by an auditor. Nevertheless, documentation often comes in the form of manually written textual documents in different formats. Moreover, the project members use diverse proprietary tools. This makes it difficult for auditors to understand how the actual project was conducted. My research addresses the project mining problem by exploiting logs from project-generated artifacts, which come from software repositories used by the project team.

Keywords: Project-Oriented Business Processes; Software Projects; Process Mining.

1 Research Problem

Companies working on human-centric projects [Ba17], such as the installation of railway interlocking systems comprising controlling software, demand for dependable and traceable processes in order to ease the verification of compliance to existing safety norms and regulations imposed by the domain of work. Usually these processes are not modeled a priori, but rather carried out in an ad-hoc manner. Furthermore, they are typically executed once with a predefined goal, but without the support of a centralized management system. This specific category of processes take the name of *project-oriented business processes*. For simplicity, we also refer to these kind of processes as *projects*. Projects do not usually make use of any software engine to support their execution, and the only way to trace them is by manual inspection of the generated artifacts, reports and logs from tools that are used by the stakeholders.

Process mining is an affirmed discipline that allows to infer a process model from log data. Nevertheless, process mining algorithms work with structured data from logs containing several process execution traces that are uniquely identified. This is not the case with projects for two reasons. On the one hand, projects are not recurrent in nature, i.e. they are executed according to a prior plan until they reach their goal, and then need to be re-planned On the other hand, projects do not rely on an execution engine that generates structured log data, but rather on proprietary tools and manually written documentation, which is often tracked by version control systemss (VCSs) and other software repositories.

This study addresses the problem of understanding projects by using the information that comes from the data recorded during their execution, with the final goal of easing auditing

[1] Vienna University of Economics and Business, saimir.bala@wu.ac.at. This work has been funded by the Austrian Research Promotion Agency (FFG) under grant 845638 (SHAPE).

and compliance checking. More specifically, the study aims at answering the question "*How can we best support project managers in understanding the as-is project by using evidence from the generated-artifacts?*". The thesis will use data from VCS logs and mailing lists in order to infer project features related to compliance, for example project activities, project phases, workload, roles, et cetera.

The rest of this paper is organized as follows. Section 2 explains the adopted research method. Section 3 describes how the work will address the identified research question. Section 4 presents preliminary results achieved so far and the next steps. Section 5 discusses related work.

2 Research Methodology

Design Science Research (DSR) is a research method which creates and evaluates IT artifacts intended to solve identified organizational problems. Figure 1 illustrates the DSR process. This research plans to design new artifacts to provide solutions to existing real world problems, following the design science process [Pe07]. More specifically, it will address the seven steps of DSR as follows.

First, after a systematic literature review of the fields of project monitoring and process mining, it will use state-of-the-art techniques and develop new artifacts to capture project information from structured and unstructured types of data. Second, it will build upon real world needs which include human-centric safety-critical automated solutions in industry (e.g., railway domain). Third, the newly designed algorithms will be converted to operational software which is an instantiated artifact [GH13] that will be tested against real data. Fourth, the contribution of this research will be positioned as a set of *exapted* methods (cf. DSR knowledge contribution framework [GH13]) from the fields of process mining and text mining, which contribute to a better understanding of projects. Fifth, the constructed artifacts will rigorously build upon state-of-the-art methods from process mining, mining software repositories and text mining, and will extend existing methods to a new problem domain, i.e. *projects*. Sixth, it will follow the design process of [Pe07]. Given the nature of the data, an exploratory phase may be required as the initial activity of this process. Seventh, it will use guidelines [GH13, RM15] in order to properly position the work. The main target will be conferences and journals related to Business Process Management (BPM) and software engineering.

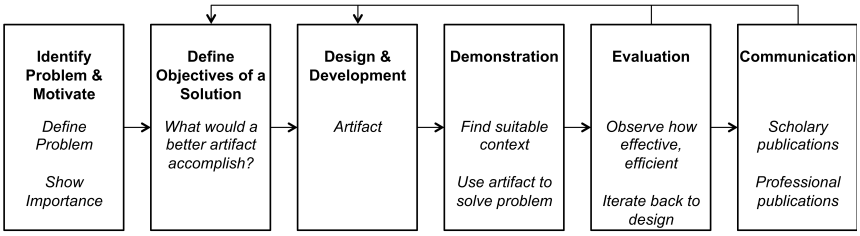

Fig. 1: Design Science Process

3 Proposed Solution

Projects are typically traced by project management tools, mailing lists, various artifacts such as word processor documents, and version control systems (VCS) which are used to manage revisions. VCSs keep track of the evolution of the generated artifacts and are used collaboratively by project members. This research aims at using VCS data to mine the actual project Gantt chart, as it is reflected by the history logs.

Given the single instance nature of projects, many process mining techniques can not be adopted directly. Although some project activities are recurrent and can be reduced to process mining problems, for example, bug fixing cycles, others are more challenging and call for other methods. The goal is to extend the field of process mining to work with unstructured data that are not generated from a business process engine, such as for instance, logs from VCS. This includes developing new techniques that allow to discover the as-is project from real data.

4 Results and Next Steps

This section summarizes three contributions that have been made towards supporting project auditing. The approaches use log data from VCSs from different open source project from GitHub[3] and from real world Subversion (SVN)or Git [TH10] event logs. Taking inspiration from the process mining field [vdA11, Do05], this works uses these kind of data for doing *project mining*. Although the main focus is project discover, it is possible to use SVN event in support of conformance checking and project enhancement. In the following, preliminary results are briefly described.

Project visualization. A first contribution has been made towards visualizing the as-is project from the data. Artifacts being created during the project's lifetime reflect the actual process that was followed. Starting from this idea and assuming that project members structure their work into dedicated folders per work-package and they systematically commit their changes, a technique to mine the actual Gantt chart out of VCS logs has been developed. The techniques takes an event log from SVN or Git and generates a Gantt chart visualization that allows for customizable levels of abstraction, from fine grained events visualization to coarser grained activities inferred from the underlying low level events. The goal is to help project managers to visually analyze what work was done in which work package. Work packages are composed of activities, which can in turn be composed by other sub-activities. Therefore, the artifact should enable for inspection with different granularity on Gantt chart. This approach is a first step towards mining projects. Main limitations are the reliance on strong assumptions on the file structure in the VCS repository and on the systematicity of the commits. This is not always the case in real projects, although many guidelinesand tutorials[4] strongly suggest that project members should follow a clear workflow involving systematic commits, in order to use VCSs effectively. Full details of the mining technique presented in this section are published in [Ba15].

[3] https://github.com/

[4] e.g. https://www.atlassian.com/git/tutorials/comparing-workflows/

Mining Resource Roles. In this work, the goal is to understand the actual roles of project members that collaborate using a VCS. Roles are decided in the project planning phase. This phase also involves the definition of project tasks and the assignment of suitable tasks to the various roles. Projects follow clear guidelines and different users are responsible for their task, to which they work locally. The setting is similar to the same as of the abovementioned contribution, i.e. a project in which project participants collaborate via a central software repository. A technique was developed in order to automatically classify users into different roles, e.g., developer, tester, technical writer, etc. This technique combines a "bag of words" dictionary and information about file types, and is able to classify commits and link the project members to predetermined roles. Clustering algorithms such as decision trees and k-means have been used. The algorithm has been tested on real data logs from industry Infinica GmbH, ProM, and Camunda, using respectively Mercurial[5], SVN and Git as VCS. The results were then validated through interviews with project members who could confirm or reject the outcome. In the worst case, more than 74% of commits where correctly classified. This approach support the main research question by showing whether the resources actually perform the tasks assigned to them in the planning phase. The full details of this work can be found in [Ag16].

Uncovering Hidden Work Dependencies. This contribution focuses on discovering hidden relations among artifacts generated as the result of executing the planned project-oriented business process. The work focuses on the particular case of software development projects which use a VCS to keep track of the file changes. A technique was developed that uses event data from VCS logs and discovers knowledge about hidden co-evolution among software artifacts, which seem apparently unrelated. Artifact evolution was recreated from the work history as times series and similarity measures were applied to understand the degree of co-evolution among the project file stories, which were also then shown as sequential business process. The technique was implemented as a prototype and used on real projects from GitHub, and allowed to discover work dependencies that beyond the classic functional ones such as interface-implementing class. Uncovering hidden relations that may exist among different parts of the project, can help to indicate the need for potential restructuring of it content following best practice guidelines, such as the good modularization principles. This work has been submitted to the BPM 2017 conference.

Next Steps. The next step is to position the topic between the project monitoring literature and empirical research on learning patterns from data, including process mining. A literature review is planned, which will asses existing project auditing and monitoring techniques and better understand how *project mining* techniques can help support project auditing and monitoring. Furthermore, user studies for an extensive evaluation of the usefulness of the developed prototypes are also planned.

5 Related Work

This research draws mainly from three fields: *i)* process mining; *ii)* text mining; and *iii)* mining software repositories. Process mining is an established field in literature and

[5] https://www.mercurial-scm.org/

allows for the discovery and conformance checking of processes from data logs [vdA11]. Nevertheless, process mining techniques require structured log data to work properly. Text mining offers a plethora of algorithms for extracting information from text data [AZ12]. Nevertheless, text mining has only recently been applied to understand business processes from text and still presents a number of challenges [MLP14]. Mining software repositories is a relatively new discipline that deals with data from software projects to distill valuable information about them [DX15]. In my research, I will use the findings from mining software repositories together with text mining algorithms, in order to understand the underlying processes or gather insights on the project.

References

[Ag16] Agrawal, Kushal; Aschauer, Michael; Thonhofer, Thomas; Bala, Saimir; Rogge-Solti, Andreas; Tomsich, Nico: Resource Classification from Version Control System Logs. In: EDOC Workshops. IEEE Computer Society, pp. 1–10, 2016.

[AZ12] Aggarwal, Charu C; Zhai, ChengXiang: Mining text data. Springer Science & Business Media, 2012.

[Ba15] Bala, Saimir; Cabanillas, Cristina; Mendling, Jan; Rogge-Solti, Andreas; Polleres, Axel: Mining Project-Oriented Business Processes. In: BPM. volume 9253 of Lecture Notes in Computer Science. Springer, pp. 425–440, 2015.

[Ba17] Bala, Saimir; Cabanillas, Cristina; Haselböck, Alois; Havur, Giray; Mendling, Jan; Polleres, Axel; Sperl, Simon; Steyskal, Simon: A Framework for Safety-Critical Process Management in Engineering Projects. In (Ceravolo, Paolo; Rinderle-Ma, Stefanie, eds): Data-Driven Process Discovery and Analysis. SIMPDA 2015. Springer International Publishing, Cham, pp. 1–27, 2017.

[Do05] van Dongen, Boudewijn F; de Medeiros, Ana Karla A; Verbeek, HMW; Weijters, AJMM; van Der Aalst, Wil MP: The ProM framework: A new era in process mining tool support. In: International Conference on Application and Theory of Petri Nets. Springer, pp. 444–454, 2005.

[DX15] Di Penta, Massimiliano; Xie, Tao: Guest editorial: special section on mining software repositories. Empir. Softw. Eng., 20(2):291–293, 2015.

[GH13] Gregor, Shirley; Hevner, Alan R: Positioning and Presenting Design Science Research for Maximum Impact. MIS Q., 37(2):337–355, 2013.

[MLP14] Mendling, Jan; Leopold, Henrik; Pittke, Fabian: 25 Challenges of Semantic Process Modeling. Int. J. Inf. Syst. Softw. Eng. Big Co., 1(1):78–94, 2014.

[Pe07] Peffers, Ken E N; Tuunanen, Tuure; Rothenberger, Marcus A. M.A.; Chatterjee, Samir: A design science research methodology for information systems research. J. Manag. Inf. Syst., 24(3):45–77, 2007.

[RM15] Recker, Jan; Mendling, Jan: The State of the Art of Business Process Management Research as Published in the BPM Conference. Bus. Inf. Syst. Eng., 58(1):1–18, 2015.

[TH10] Torvalds, Linus; Hamano, Junio: Git: Fast version control system. URL http://git-scm.com, 2010.

[vdA11] van der Aalst, Wil: Process mining: discovery, conformance and enhancement of business processes. Springer Science & Business Media, 2011.