Data & Knowledge Engineering 98 (2015) 144–164

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Improving business process intelligence by observing object state transitions

Nico Herzberg *, Andreas Meyer, Mathias Weske

Hasso Plattner Institute at the University of Potsdam Prof.-Dr.-Helmert-Str. 2–3, D-14482, Potsdam, Germany

ARTICLE INFO

Available online 10 July 2015

Keywords:
Business process management
Events
Data
Process Monitoring
BPMN

ABSTRACT

During the execution of business processes several events happen that are recorded in the company's information systems. These events deliver insights into process executions so that process monitoring and analysis can be performed resulting, for instance, in prediction of upcoming process steps or the analysis of the run time of single steps. While event capturing is trivial when a process engine with integrated logging capabilities is used, manual process execution environments do not provide automatic logging of events, so that typically external devices, like bar code scanners, have to be used. As experience shows, these manual steps are error-prone and induce additional work. Therefore, we use object state transitions as additional monitoring information, so-called object state transition events. Based on these object state transition events, we reason about the enablement and termination of activities and provide the basis for process monitoring and analysis in terms of a large event log. In this paper, we present the concept to utilize information from these object state transition events for capturing process progress. Furthermore, we discuss a methodology to create the required design time artifacts that then are used for monitoring at run time. In a proof-of-concept implementation, we show how the design time and run time side work and prove applicability of the introduced concept of object state transition events.

1. Introduction

Nowadays, companies face a very competitive and fast changing market environment. Therefore, they strive to run their value generating operations in a process-oriented way to keep the pace. These business processes are managed by using techniques and methodologies of business process management (BPM). BPM deals with the organization, documentation, analysis, optimization, and execution of business processes [1]. One important aspect of BPM is business process intelligence (BPI) that comprises process analysis, monitoring, and mining [2,3]. For using BPI approaches, information about process behavior and events that occur during process execution need to be present in a certain quantity and quality. Otherwise, the derived conclusion might not correspond to the actual execution.

Process monitoring is used to predict upcoming events and process steps by observing the actual process execution and deriving the corresponding process behavior. Process monitoring is usually applied on process models being enacted by a process engine – an information system that controls the process execution – because it generally provides logging capabilities out of the box. Therewith, the current process progress is easily recognizable from the observed events.

The observation of such an event determines the current position in the course of process execution. In contrast, manually executed processes, as, for instance, usual in healthcare, lack these capabilities such that most events cannot be observed or stored, i.e., event logs do not exist or are incomplete. However, recognition and prediction of process progress as well as a holistic view on the process

* Corresponding author.
E-mail addresses: nico.herzberg@hpi.de (N. Herzberg), andreas.meyer@hpi.de (A. Meyer), mathias.weske@hpi.de (M. Weske).

http://dx.doi.org/10.1016/j.datak.2015.07.008
0169-023X/

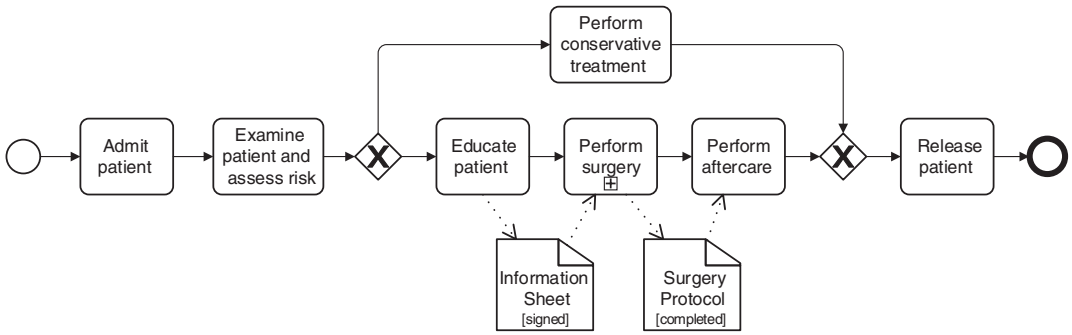

Fig. 1. Abstracted treatment process in a hospital, starting with the admission of the patient, followed by its examination and the treatment alternatives, i.e., conservative treatment and surgery, and ending with the release of the patient.

execution is necessary in these domains as well. They also need to know exactly what is going on, how the process performs, and where the points of improvements are. The lack of events could be fixed by introducing additional external devices and similar logging equipment. However, it is not applicable in manual processes – especially in time critical ones as in healthcare – due to extra amount of work and complexity. Therefore, event monitoring points [4] are defined to specify where in the process events are expected. This event information is than extracted from the IT system landscape and correlated to the particular process execution. Event monitoring points can be considered as milestones indicating the achievement of important business value.

In this paper, we apply the concept of event monitoring points to data objects to create more expressive event logs without adding additional logging mechanisms or devices such as scanners, RFID tags, or sensors. We further elaborate on the ideas about the concept of *object state transition events* (in short: *transition events*), which have been initially sketched in Ref. [5] and discussed in more detail in Ref. [6][1]. Thereby, we reason about activity termination (enablement) by means of events indicating the data objects to be written (read) by that activity transitioned into the respective object state and vice versa. Additionally, this technique also increases the number of events observed in enacted process models such that analysis is based on a much higher number of inputs resulting in an increased reliability of the results.

The remainder of the paper is structured as follows. We present a motivating example from the healthcare domain and discuss the arising challenges in Section 2. In Section 3, we introduce the foundation combined with the scope of our approach followed by an approach overview and detailed descriptions of its application in Section 4. The general methodology to apply the presented approach is defined in Section 5. Section 6 describes the application of the introduced approach to a use case during *design time* (model view) and during *run time* (instance view). In Section 7, the system architecture including an implementation is described, followed by an evaluation of the approach by applying the use case from Section 6 to the implemented software system. Section 8 discusses the additional benefits of the presented approach for BPI application in non-automated process execution environments. Finally, we discuss related work in Section 9 and conclude the paper in Section 10.

2. Motivating example

Within healthcare, business processes are executed by humans in a manual fashion. Some steps are possibly connected with IT systems, e.g., the admission and release of a hospital's patient. During patient treatment several information are taken to paper and various forms are filled and brought into document systems later on.

In Fig. 1, an abstracted treatment process is shown. The process starts with the admission of the patient, followed by its examination and the risk assessment for a particular treatment. Based on that, the decision between a conservative treatment and a surgery is made. In case of a surgery, the patient must be educated beforehand and needs aftercare afterwards. In any case, the patient is released after the treatment. In this process, two documents are of importance, the *Information Sheet* and the *Surgery Protocol*. These documents are valuable, because they are the manipulated artifacts during process execution and describe the processing results through their changing states. For instance, a *Surgery Protocol* has several parts like patient data, staff data, medical data, and surgery data. Filling those parts put the document in a certain state, e.g., *patient data filled* representing completeness of this respective part. The document *Information Sheet* is, for instance, described through the states *initial*, *filled*, and *signed*. During process execution, the *Information Sheet* could be in one of these states at some point in time but in no other.

In this scenario, only the first activity *Admit patient* and the last activity *Release patient* describe work that is done in an IT system. Thus, the usual log of recorded activities for a particular process execution only contains events for these activities and is as such not efficiently usable for typical BPI applications, e.g., process mining [3] and business process monitoring [7]. Therewith, this process environment challenges the requirement for business process monitoring, as (a) several steps of the patients treatment are not represented in IT systems – in fact, only the admission and the release of a patient are –, (b) there is no central system to collect

[1] This paper at hand is a special issue paper extending the paper [6] published through ER'13 conference.