Auditing 2.0: Using Process Mining to Support Tomorrow's Auditor

Wil M.P. van der Aalst, Eindhoven University of Technology & Queensland University of Technology
Kees M. van Hee, Eindhoven University of Technology
Jan Martijn van der Werf, Eindhoven University of Technology
Marc Verdonk, Deloitte Netherlands & Eindhoven University of Technology

The term *auditing* refers to the evaluation of organizations and their processes. Audits are performed to ascertain the validity and reliability of information about these organizations and associated processes. This is done to check whether business processes are executed within certain *boundaries* set by managers, governments, and other stakeholders. For example, specific rules may be enforced by law or company policies and the auditor should check whether these rules are followed or not. Violations of these rules may indicate fraud, malpractice, risks, and inefficiencies. Traditionally, an auditor can only provide reasonable assurance that business processes are executed within the given set of boundaries. They check the operating effectiveness of controls that are designed to ensure reliable processing. When these controls are not in place, or otherwise not functioning as expected, they typically only check samples of factual data, often in the 'paper world'. However, today detailed information about processes is being recorded in the form of event logs, audit trails, transaction logs, databases, data warehouses, etc. Therefore, it should no longer be necessary to only check a small set of samples offline. Instead, all events in a business process can be evaluated and this can be done while the process is still running. The availability of log data and *advanced process mining techniques* enable a new form of auditing: *Auditing 2.0*. Surely, the availability of process mining techniques and the omnipresence of recorded business events will dramatically change the role of auditors.

PROCESS MINING

The idea of process mining is to discover, monitor and improve real processes (i.e., not assumed processes) by extracting knowledge from event logs. Over the last decade process mining techniques have matured and are currently being integrated in commercial software products (W.M.P. van der Aalst, H.A. Reijers, A.J.M.M. Weijters, B.F. van Dongen, A.K. Alves de Medeiros, M. Song, and H.M.W. Verbeek. Business Process Mining: An Industrial Application. Information Systems, 32(5):713-732, 2007).

Business provenance

Starting point for process mining are so-called *event logs*, i.e., sequentially recorded collections of events such that each event refers to an *activity* (i.e., a well-defined step in the process) and is related to a particular *case* (i.e., a process instance). Furthermore, some mining techniques use additional information such as the performer or originator of the event (i.e., the person/resource executing or initiating the activity), the timestamp of the event, or data elements recorded with the event (e.g., the size of an order). Note that from an auditing point of view the systematic, reliable, and trustworthy recording of events is essential. This is sometimes referred to as *business provenance*. This term acknowledges the importance of traceability by making sure that "history cannot be rewritten or obscured".

Process discovery

Using process mining techniques it is possible to discover processes. Based on event logs models can be extracted that describe the processes and organizations at hand. In this case, there is no a-priori handcrafted model; it is learned by analyzing frequent patterns. For example, well-known algorithms such as the Alpha algorithm can automatically extract a Petri net that gives a concise model of the behavior seen in the event log. This gives the auditor an unbiased view on what has actually happened.

Conformance checking

If there is an a-priori model, then this model can be used to check if reality, as recorded in the log, conforms to the model and vice versa. For example, there may be a process model indicating that purchase orders of more than one million Euros require two checks. Another example is the checking of the four-eye principle. Conformance checking may be used to detect deviations, to locate and explain these deviations, and to measure the severity of these deviations. An example is the conformance checking algorithm described in (A. Rozinat and W.M.P. van der Aalst. Conformance Checking of Processes Based on Monitoring Real Behavior. *Information Systems*, 33(1):64-95, 2008).

Model extension

An a-priori model can also be extended with a new aspect or perspective based on log data. The goal is not to check conformance but to enrich the model. An example is the extension of a process model with performance data, e.g. to find bottlenecks in a process model.

Towards operational support

Traditionally, process mining has been focusing on off-line analysis and is seldom used for operational decision support. However, it is also possible to use process mining in an online fashion by considering events of running process instances and comparing them with models based on historic data or business rules (W.M.P. van der Aalst, M. Pesic, and M. Song. Beyond Process Mining: From the Past to Present and Future. *BPM Center Report* BPM-09-18, BPMcenter.org, 2009). For example, one can "replay" a running case on the process model at real-time and check whether the observed behavior fits. The moment the case deviates, an appropriate actor can be alerted. The process model based on historic data can also be used to make predictions for running cases, e.g., it is possible to estimate the remaining processing time and the probability of a particular outcome. Similarly, this information can be used to provide recommendations, e.g., proposing the activity that will minimize the expected costs and completion time.

AUDITING FRAMEWORK

The presence of event logs and process mining techniques enables new forms of auditing. Rather than sampling a small set of cases, the whole process and all of its instances can be considered. Moreover, this can be done continuously. The auditing framework shown in Figure 1 illustrates the ideas behind

Auditing 2.0. Information is recorded in events logs: data store *current data* refers to events relating to cases that are still running and data store *historic data* refers to events of cases that have completed. Figure 1 also shows two types of models: *de jure models* are normative models that describe a desired or required way of working while *de facto models* aim to describe reality with potential violations of the boundaries described in de jure models (W.M.P. van der Aalst, K.M. van Hee, J.M. van der Werf, A. Kumar, and M. Verdonk. Conceptual Model for On Line Auditing. *BPM Center Report* BPM-09-19, BPMcenter.org, 2009).

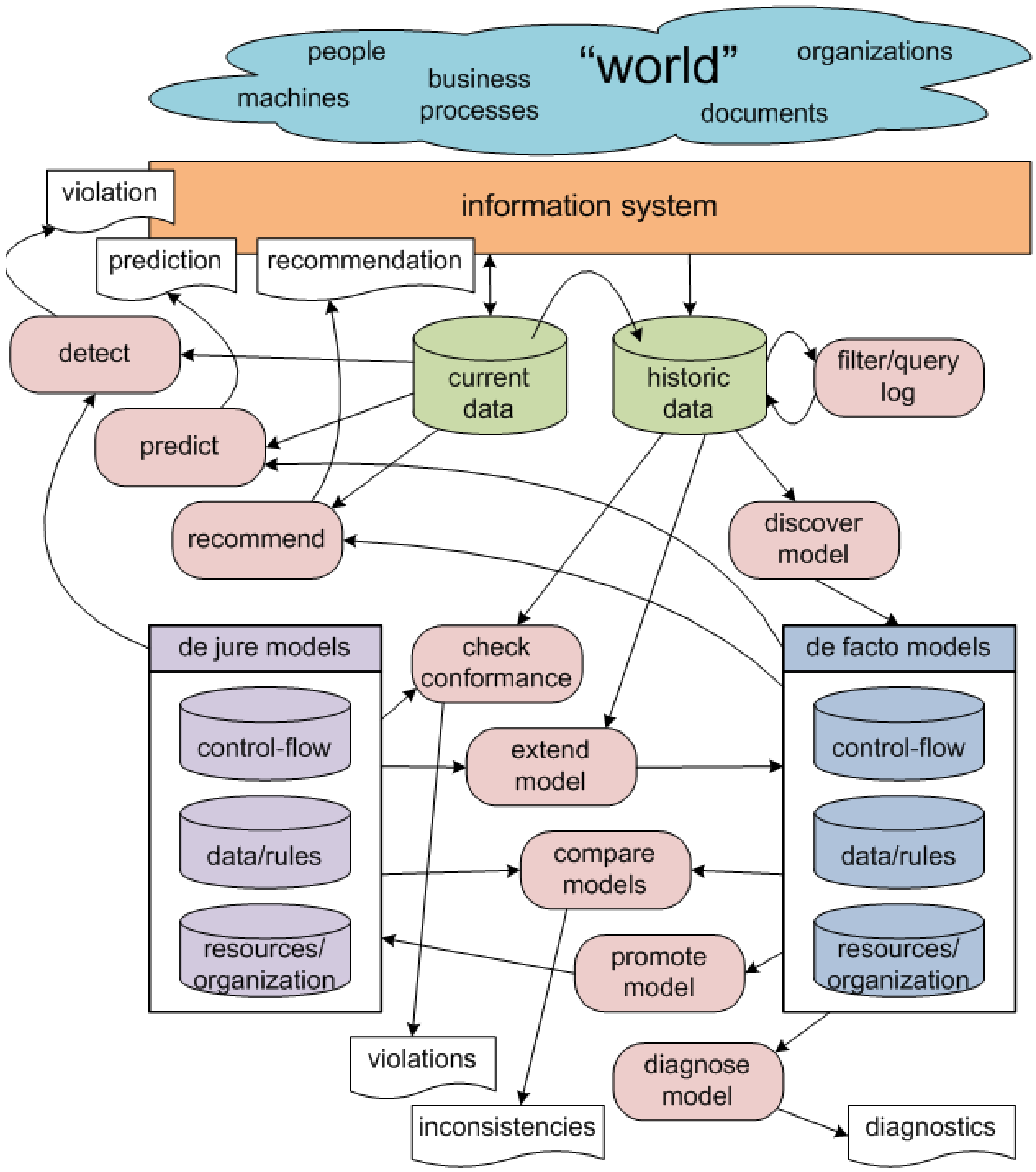

Figure 1: Auditing framework based on process mining. Information about the "world" is recorded in event logs and related to "de jure" models and "de facto" models.

Auditing using historic data

Historic data, i.e., event logs of cases that have completed, play a valuable role in off-line auditing. As Figure 1 shows, historic data can be used to *filter and query the log*. Filtering may be needed to remove irrelevant situations or to scope the event log (e.g., for a particular process or group of customers). While filtering, entire cases can be removed (e.g. remove all process instances related to gold customers in a particular region) or individual events can be removed (e.g. remove all checking events done by people from a particular department). The result is a smaller, better scoped, event log that can be used for further analysis. Querying the log can be done to manually search for particular cases or events. This is particularly useful in case of ad-hoc auditing questions.

Historic data in the form of event logs can also be used to *discover de facto models*. These models can cover different perspectives, e.g., control-flow (the ordering of activities), data/rules, and resources/organization. Note that most process mining algorithms focus on process discovery with an emphasis on the control flow, cf. the Alpha algorithm mentioned earlier that is able to automatically extract a Petri net model explaining the recorded history. However, there are also process mining algorithms to discover organizational models and classical data mining algorithms such as ID3, C4.5 and CART are able to extract decision trees based on data attributes.

Historic data can also be analyzed with respect to *de jure models*. *Conformance checking* techniques pinpoint deviations, e.g., highlighting parts of the model were the conformance is low or pointing out cases that deviate. This is highly relevant for auditing purposes. It can be used to see which rules are violated and where and when people do not execute processes as specified.

Finally, historic data can be used to extend existing models. For example, service levels and other performance indicators can be measured and projected onto the model. This provides the auditor with diagnostic information that can be used to spot possible problems.

Auditing using models only

The lower part of Figure 1 shows various types of analysis that do not directly involve any event data. First of all, it is possible to *compare de jure models and de facto models* and analyze the differences and commonalities. For example, if a de facto process model obtained using process discovery shows paths that are not possible according to the de jure model, then this serves as a good starting point for an in-depth analysis by the auditor.

Secondly, it is possible to *promote a de facto model to become a de jure model*. If comparison shows that the actual way of working is not consistent with the normal pre-existing model, this may be a reason to update the de jure model. If people find better ways to execute processes, then this may be adopted as the “new way of working”.

Finally, the de facto models may be diagnosed using conventional model-based analysis techniques. For example, models can be checked for deadlocks and other anomalies. Moreover, process mining results can be merged to discover comprehensive simulation models incorporating the various perspectives. These models can be used for what-if analysis.

Auditing using current data

Classically, the role of an auditor is to check *afterwards* whether the business processes where executed within certain boundaries. However, the omnipresence of real-time event data and the capabilities of today's IT systems, make it possible to monitor processes on-the-fly. Note that the data store *current data* in Figure 1 holds events related to cases that are still running. These become *historic data* when they complete. However, before completion, when it is still possible to influence the operational process, actions can be triggered by the auditor. Recently, various process mining techniques for operational support emerged. These are typically based on "replay", e.g., by playing the well-known "token game" for Petri nets in a smart way, one can detect deviations, predict particular outcomes, and recommend appropriate actions. Note that it is possible to map business rules onto Petri nets or temporal logic (e.g. LTL) thus enabling efficient checks.

By comparing the information about running cases with the de jure model, deviations can be *detected* as they occur. In fact it is possible to *predict* whether deviations are likely to occur. Consider for example a legal deadline such as "Claims need to be handled within three weeks". Various techniques can be used signal the likelihood of violating such a deadline (W.M.P. van der Aalst, M.H. Schonenberg, and M. Song. Time Prediction Based on Process Mining. *BPM Center Report* BPM-09-04, BPMcenter.org, 2009). Similar techniques can be used to *recommend* particular actions, e.g., "Taking action X will minimize the risk of violating legal requirement Y".

The possibility to provide operational support creates an interesting dilemma. On the one hand, it seems odd not to act based on information that is readily available. On the other hand, the auditor may lose its independence by interfering with the operational process. For example, if an auditor provides warnings while the process is still running, she becomes partially involved in the actual execution. Can she still assess the process afterwards?

PROM AS AN AUDITING PLATFORM

The *ProM toolset* (www.processmining.org) aims to operationalize the framework described in Figure 1. Several parts of our auditing framework have been implemented. ProM has a pluggable architecture and supports a wide range of control-flow models, e.g., various types of Petri nets, EPCs, BPMN, BPEL, etc. Also models to represent rules (e.g., LTL-based), social networks, and organizational structures are supported. Moreover, for each of the activities shown in Figure 1 there are multiple plug-ins available. For example, there are dozens of plug-ins to discover the various types of models supported by ProM, there are also various ways of checking the conformance of process models, and more recently plug-ins have been added for operational support (i.e., supporting the *detect*, *predict*, and *recommend* activities in Figure 1).

ProM is open source and can be downloaded from *www.processmining.org*. Figure 2 shows a screenshot of ProM while checking the conformance of process in a Dutch municipality. Although ProM serves as an excellent basis for Auditing 2.0, it is not yet tailored towards the specific needs of auditors. ProM is a generic multi-purpose tool, so the aim is to develop a customized version of ProM based on the framework described in Figure 1.

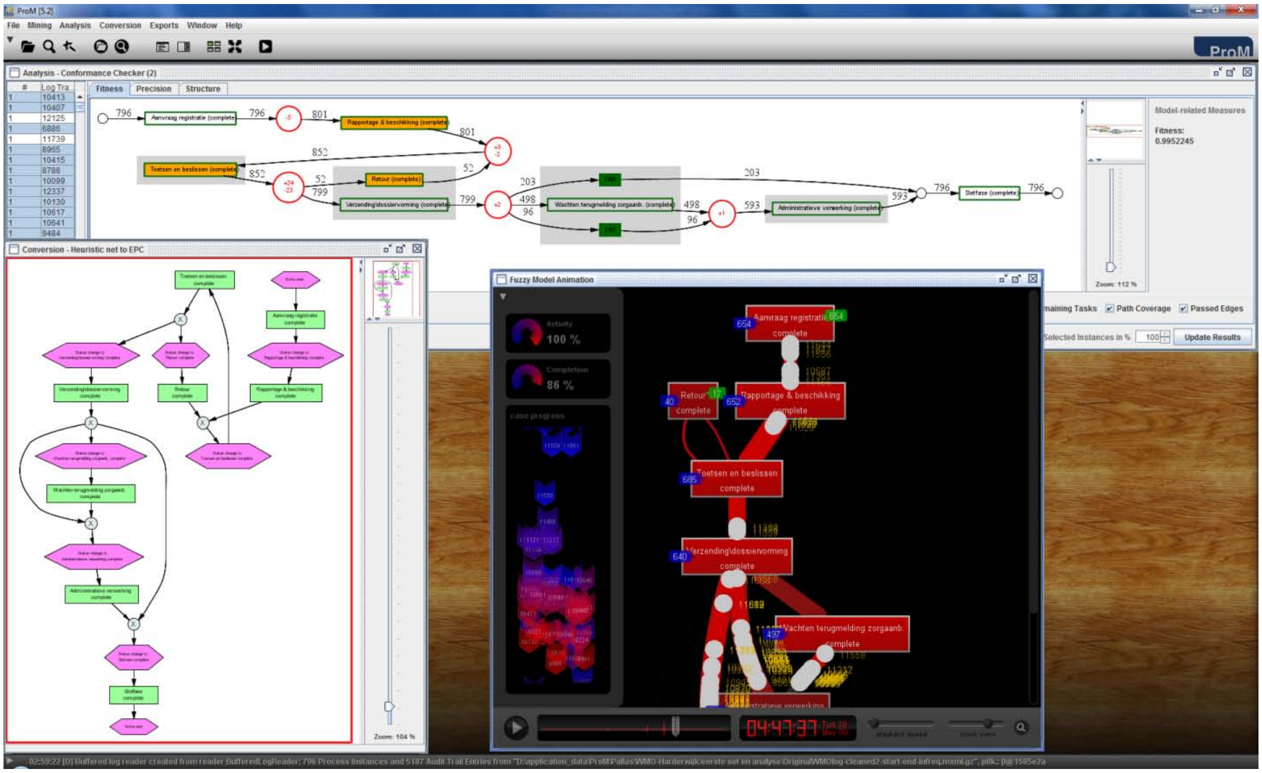

Figure 2: Screenshot of ProM while analyzing the conformance of a process inside a Dutch municipality based on an event log containing 5187 events related to 796 cases (applications for support by citizens). Analysis shows the overall conformance (99.5 percent) and highlights the parts of the process where deviations are most frequent.

Challenges

The application of process mining to auditing depends first and foremost on the availability of relevant data. This data is primarily stored in ERP systems like SAP. Mining ERP-systems is challenging because these systems are not process-oriented (despite built-in workflow engines) and data related to a particular process is typically scattered over dozens of tables. Hence, it is a non-trivial exercise to extract minable auditing data.

A second challenge concerns the current practice of auditing. Driven by the so-called "auditing materiality" principle, the auditor typically considers only a small subset of data. If no deviations are seen, no further actions are needed. By looking at all the data, the auditor inevitably finds more exceptions to follow up, arguably increasing quality, but also increasing the time (and cost) of the audit. Moreover, for a widespread adoption of process mining as an accepted auditing approach, the methodologies and guidelines issued by organizations such as IFAC (International Federation of Accountants) have to be changed, as companies need to rely on these, e.g., for insurance reasons.

OUTLOOK

Major corporate and accounting scandals including those affecting Enron, Tyco, Adelphia, Peregrine and WorldCom have fueled the interest in more rigorous auditing practices. Legislation such as the Sarbanes–

Oxley (SOX) Act of 2002 and the Basel II Accord of 2004 was enacted as a reaction to such scandals. Also the recent financial crisis emphasizes the importance of verification whether organizations operate "within their boundaries". Process mining techniques offer a means to more rigorously check compliance and ascertain the validity and reliability of information about an organization's core processes.

Auditing 2.0 - a more rigorous form of auditing based on detailed event logs while using process mining techniques - will change the job description of tomorrow's auditor dramatically. Auditors will be required to have better analytical and IT skills and their role will shift as auditing is done on-the-fly. Moreover, more emphasis will be put on the recording of business events as is reflected by the term "business provenance". Based on provenance data in the form of event logs, it should be possible to replay history reliably and accurately. Moreover, by having high-quality event logs, process mining techniques can be used to improve business processes and predict problems.

Wil M.P. van der Aalst is a full professor of Information Systems in the Mathematics and Computer Science Department at Eindhoven University of Technology where leads the Architecture of Information Systems (AIS) group. He is also adjoint professor at Queensland University of Technology. Contact him at w.m.p.v.d.aalst@tue.nl.

Kees M. van Hee is also a full professor in the AIS group at Eindhoven University of Technology. Contact him at k.m.v.hee@tue.nl.

Jan Martijn van der Werf is a PhD candidate in the AIS group at Eindhoven University of Technology. Contact him at j.m.e.m.v.d.werf@tue.nl.

Marc Verdonk is a senior manager and IT auditor at Deloitte Enterprise Risk Services also doing a PhD in the AIS group at TU/e. Contact him at mverdonk@deloitte.nl.