Process Mining: X-Ray Your Business Processes

WIL VAN DER AALST, Eindhoven University of Technology

Recent breakthroughs in *process mining* research make it possible to discover, analyze, and improve business processes based on event data. Activities executed by people, machines, and software leave trails in so-called *event logs*. Events such as entering a customer order into SAP, checking in for a flight, changing the dosage for a patient, and rejecting a building permit have in common that they are all recorded by information systems. Over the last decade there has been a spectacular growth of data. Moreover, the digital universe and the physical universe are becoming more and more aligned. Therefore, business processes should be managed, supported, and improved based on event data rather than subjective opinions or obsolete experiences. The application of process mining in hundreds of organizations has shown that both managers and users tend to overestimate their knowledge of the processes they are involved in. Hence, process mining results can be viewed as X-rays showing what is *really* going on *inside* processes. Such X-rays can be used to diagnose problems and suggest proper treatment. The practical relevance of process mining and the interesting scientific challenges make process mining one of the "hot" topics in Business Process Management (BPM). This article provides an introduction to process mining by explaining the core concepts and discussing various applications of this emerging technology.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms: Management, Measurement, Performance

Additional Key Words and Phrases: Process mining, business intelligence, business process management, data mining

ACM Reference Format:
Van der Aalst, W.M.P. 2011. Process Mining: X-Ray Your Business Processes. CACM 99, 99, Article 99 (November 2011), 11 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. PROCESS MINING SPECTRUM

Process mining aims to *discover, monitor and improve real processes by extracting knowledge from event logs* readily available in today's information systems [Aalst 2011a; 2011b]. Although event data are omnipresent, organizations lack a good understanding of their actual processes. Management decisions tend to be based on PowerPoint diagrams, local politics, or management dashboards rather than an careful analysis of event data. The knowledge hidden in event logs cannot be turned into actionable information. Advances in data mining made it possible to find valuable patterns in large datasets and to support complex decisions based on such data. However, classical data mining problems such as classification, clustering, regression, association rule learning, and sequence/episode mining are *not* process-centric. Therefore, Business Process Management (BPM) approaches tend to resort to hand-made mod-

Author's address: Department of Mathematics and Computer Science, Eindhoven University of Technology, PO Box 513, 5600 MB, Eindhoven, The Netherlands. `w.m.p.v.d.aalst@tue.nl`

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

els. Process mining research aims to bridge the gap between data mining and BPM. Metaphorically, process mining can be seen as taking X-rays to diagnose/predict problems and recommend treatment.

An important driver for process mining is the incredible growth of event data [Hilbert and Lopez 2011; Manyika et al. 2011]. Event data is everywhere – in every sector, in every economy, in every organization, and in every home one can find systems that log events. For less than $600, one can buy a disk drive with the capacity to store all of the world's music [Manyika et al. 2011]. A recent study published in *Science* [Hilbert and Lopez 2011], shows that storage space grew from 2.6 optimally compressed exabytes (2.6×10^{18} bytes) in 1986 to 295 compressed exabytes in 2007. In 2007, 94 percent of all information storage capacity on Earth was digital. The other 6 percent resided in books, magazines and other non-digital formats. This is in stark contrast with 1986 when only 0.8 percent of all information storage capacity was digital. These numbers illustrate the exponential growth of data.

The further adoption of technologies such as RFID (Radio Frequency Identification), location-based services, cloud computing, and sensor networks, will further accelerate the growth of event data. However, organizations have problems effectively using such large amounts of event data. In fact, most organizations still diagnose problems based on fiction (Powerpoint slides, Visio diagrams, etc.) rather than facts (event data). This is illustrated by the poor quality of process models in practice, e.g., more than 20% of the 604 process diagrams in SAP's reference model have obvious errors and their relation to the actual business processes supported by SAP is unclear [Mendling et al. 2007]. Therefore, it is vital to turn the massive amounts of event data into relevant knowledge and reliable insights. This is where process mining can help.

The growing maturity of process mining is illustrated by the *Process Mining Manifesto* [TFPM 2012] recently released by the *IEEE Task Force on Process Mining*. This manifesto is supported by 53 organizations and 77 process mining experts contributed to it. The active contributions from end-users, tool vendors, consultants, analysts, and researchers illustrate the significance of process mining as a bridge between data mining and business process modeling.

Starting point for process mining is an *event log*. Each event in such a log refers to an *activity* (i.e., a well-defined step in some process) and is related to a particular *case* (i.e., a *process instance*). The events belonging to a case are *ordered* and can be seen as one "run" of the process. Event logs may store additional information about events. In fact, whenever possible, process mining techniques use extra information such as the *resource* (i.e., person or device) executing or initiating the activity, the *timestamp* of the event, or *data elements* recorded with the event (e.g., the size of an order).

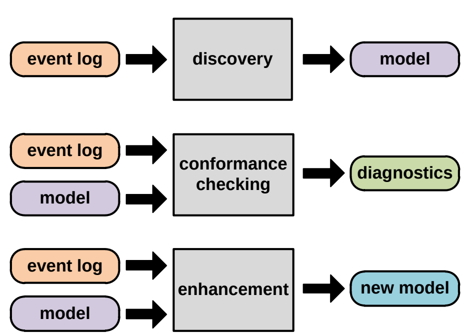

Fig. 1. The three basic types of process mining explained in terms of input and output.

Event logs can be used to conduct three types of process mining as shown in Fig. 1 [Aalst 2011a]. The first type of process mining is *discovery*. A discovery technique takes an event log and produces a model without using any a-priori information. Process discovery is the most prominent process mining technique. For many organizations it is surprising to see that existing techniques are indeed able to discover real processes merely based on example behaviors recorded in event logs. The second type of process mining is *conformance*. Here, an existing process model is compared with an event log of the same process. Conformance checking can be used to check if reality, as recorded in the log, conforms to the model and vice versa. The third type of process mining is *enhancement*. Here, the idea is to extend or improve an existing process model using information about the actual process recorded in some event log. Whereas conformance checking measures the alignment between model and reality, this third type of process mining aims at changing or extending the a-priori model. For instance, by using timestamps in the event log one can extend the model to show bottlenecks, service levels, throughput times, and frequencies.

2. PROCESS DISCOVERY

As shown in Fig. 1, the goal of process discovery is to learn a model based on some event log. Events can have all kinds of attributes (timestamps, transactional information, resource usage, etc.). These can all be used for process discovery. However, for simplicity, we often represent events by activity names only. This way, a case (i.e., process instance) can be represented by a *trace* describing a sequence of activities. Consider for example the event log shown in Fig. 2 (example is taken from [Aalst 2011a]). This event log contains 1391 cases, i.e., instances of some reimbursement process. There are 455 process instances following trace $acdeh$. Activities are represented by a single character: a = *register request*, b = *examine thoroughly*, c = *examine casually*, d = *check ticket*, e = *decide*, f = *reinitiate request*, g = *pay compensation*, and h = *reject request*. Hence, trace $acdeh$ models a reimbursement request that was rejected after a registration, examination, check, and decision step. 455 cases followed this path consisting of five steps, i.e., the first line in the table corresponds to $455 \times 5 = 2275$ events. The whole log consists of 7539 events.

Process discovery techniques produce process models based on event logs such as the one shown in Fig. 2. For example, the classical α-algorithm produces model M_1 for this log. This process model is represented as a *Petri net*. A Petri net consists of *places* and *transitions*. The state of a Petri net, also referred to as *marking*, is defined by the distribution of *tokens* over places. A transition is *enabled* if each of its input places contains a token. For example, a is enabled in the initial marking of M_1, because the only input place of a contains a token (black dot). Transition e in M_1 is only enabled if both input places contain a token. An enabled transition may *fire* thereby consuming a token from each of its input places and producing a token for each of its output places. Firing a in the initial marking corresponds to removing one token from *start* and producing two tokens (one for each output place). After firing a, three transitions are enabled: b, c, and d. Firing b will disable c because the token is removed from the shared input place (and vice versa). Transition d is concurrent with b and c, i.e., it can fire without disabling another transition. Transition e becomes enabled after d and b or c have occurred. After executing e three transitions become enabled: f, g, and h. These transitions are competing for the same token thus modeling a choice. When g or h is fired, the process ends with a token in place *end*. If f is fired, the process returns to the state just after executing a.

Note that transition d is concurrent with b and c. Process mining techniques need to be able to discover such more advanced process patterns and should not be restricted to simple sequential processes.

event log

#	trace
455	acdeh
191	abdeg
177	adceh
144	abdeh
111	acdeg
82	adceg
56	adbeh
47	acdefdbeh
38	adbeg
33	acdefbdeh
14	acdefbdeg
11	acdefdbeg
9	adcefcdeh
8	adcefdbeh
5	adcefbdeg
3	acdefbdefdbeg
2	adcefdbeg
2	adcefbdefbdeg
1	adcefdbefbdeh
1	adbefbdefdbeg
1	adcefdbefcdefdbeg
1391	

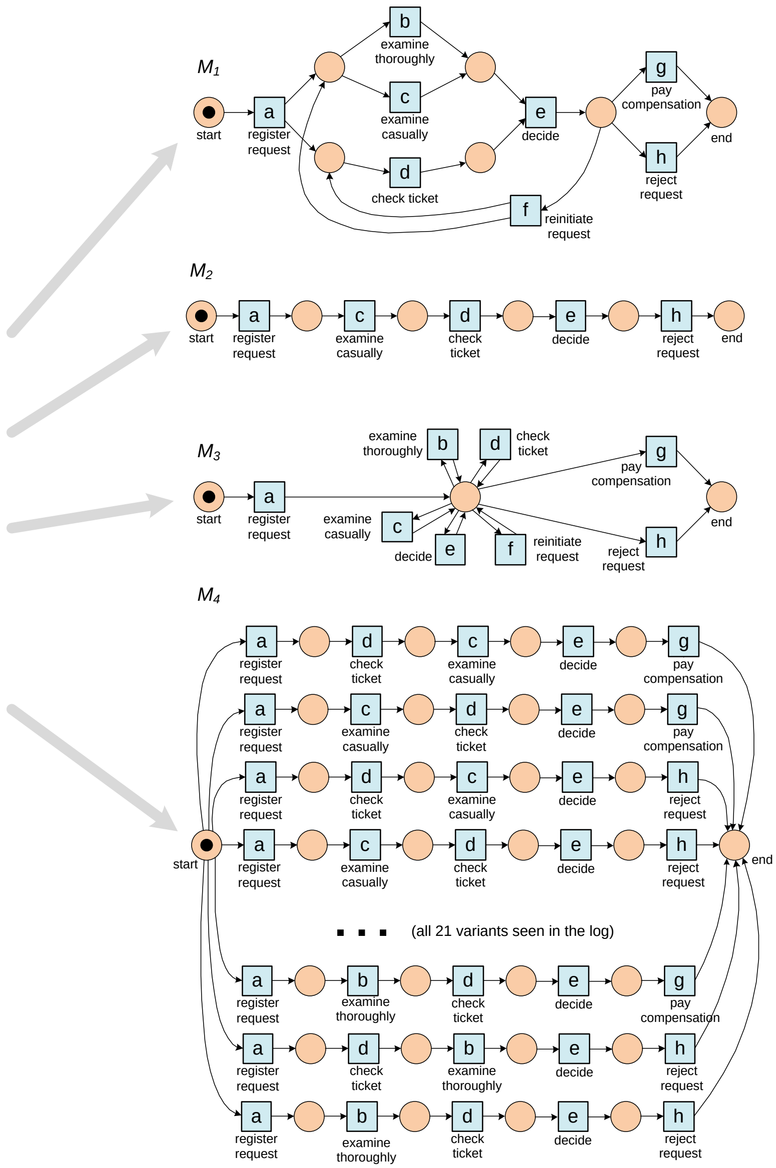

Fig. 2. One event log and four potential process models (M_1, M_2, M_3, and M_4) aiming to describe the observed behavior.

It is easy to check that all traces in the event log can be reproduced by M_1. This does not hold for the second process model in Fig. 2. M_2 is only able to reproduce the most frequent trace $acdeh$. The model does not *fit* the log well because observed traces such as $abdeg$ are not possible according to M_2. The third model is able to reproduce the entire event log, but M_3 also allows for traces such as ah and $adddddddg$. Therefore, we consider M_3 to be "underfitting"; too much behavior is allowed because M_3 clearly overgeneralizes the observed behavior. Model M_4 is also able to reproduce the event log. However, the model simply encodes the example traces in the log. We call such a model "overfitting" as the model does not generalize behavior beyond the observed examples.

In recent years, powerful process mining techniques have been developed that can automatically construct a suitable process model given an event log. The goal of such techniques is to construct a simple model that is able to explain most of the observed behavior without "overfitting" or "underfitting" the log.

3. CONFORMANCE CHECKING

Process mining is not limited to process discovery. In fact, the discovered process is merely the starting point for deeper analysis. As shown in Fig. 1, conformance checking and enhancement relate model and log. The model may have been made by hand or discovered through process discovery. For conformance checking, the modeled behavior and the observed behavior (i.e., event log) are compared. When checking the conformance of M_2 with respect to the log shown in Fig. 2, it is easy to see that only the 455 cases that followed $acdeh$ can be replayed from begin to end. If we try to replay trace $acdeg$, we get stuck after executing $acde$ because g is not enabled. If we try to replay trace $adceh$, we get stuck after executing the first step because d is not (yet) enabled.

There are various approaches to diagnose and quantify conformance. One approach is to find an *optimal alignment* between each trace in the log and the most similar behavior in the model. Consider for example process model M_1, a fitting trace $\sigma_1 = adceg$, a non-fitting trace $\sigma_2 = abefdeg$, and the following three alignments:

$$\gamma_1 = \begin{array}{|c|c|c|c|c|} a & d & c & e & g \\ \hline a & d & c & e & g \end{array} \quad \text{and} \quad \gamma_2 = \begin{array}{|c|c|c|c|c|c|c|c|c|} a & b & \gg & e & f & d & \gg & e & g \\ \hline a & b & d & e & f & d & b & e & g \end{array} \quad \text{and} \quad \gamma_3 = \begin{array}{|c|c|c|c|c|c|c|} a & b & e & f & d & e & g \\ \hline a & b & \gg & \gg & d & e & g \end{array}$$

γ_1 shows a perfect alignment between σ_1 and M_1: all moves of the trace in the event log (top part of alignment) can be followed by moves of the model (bottom part of alignment). γ_2 shows an optimal alignment for trace σ_2 in the event log and model M_1. The first two moves of the trace in the event log can be followed by the model. However, e is not enabled after executing just a and b. In the third position of alignment γ_2, we see a d move of the model that is not synchronized with a move in the event log. A move in just the model is denoted as $(\gg, d)$. In the next three moves model and log agree. In the seventh position of alignment γ_2 there is just a move of the model and not a move in the log: $(\gg, b)$. γ_3 shows another optimal alignment for trace σ_2. Here there are two situations where log and model do not move together: $(e, \gg)$ and $(f, \gg)$. Alignments γ_2 and γ_3 are both optimal if the penalties for "move in log" and "move in model" are the same. In both alignments there are two $\gg$ steps and there are no alignments with less than two $\gg$ steps.

Conformance can be viewed from two angles: (a) the model does not capture the real behavior ("the model is wrong") and (b) reality deviates from the desired model "the event log is wrong"). The first viewpoint is taken when the model is supposed to be *descriptive*, i.e., capture or predict reality. The second viewpoint is taken when the model is *normative*, i.e., used to influence or control reality.

There are various types of conformance and creating an alignment between log and model is just the starting point for conformance checking [Aalst 2011a]. For example, there are various *fitness* (the ability to replay) metrics. A model has fitness 1 if all traces can be replayed from begin to end. A model has fitness 0 if model and event log "disagree" on all events. Process models M_1, M_3 and M_4 have a fitness of 1 (i.e., perfect fitness) with respect to the event log shown in Fig. 2. Model M_2 has a fitness 0.8 for the event log consisting of 1391 cases. Intuitively, this means that 80% of the events in the log can be explained by the model. Fitness is just one of several conformance metrics.

Experiences with conformance checking in dozens of organizations show that real-life processes often deviate from the simplified Visio or PowerPoint representations used by process analysts.

4. MODEL ENHANCEMENT

It is also possible to extend or improve an existing process model using the alignment between event log and model. A non-fitting process model can be corrected using the diagnostics provided by the alignment. If the alignment contains many $(e, \gg)$ moves, then it may make sense to allow for the skipping of activity e in the model. Moreover, event logs may contain information about resources, timestamps, and case data. For example, an event referring to activity "register request" and case "992564" may also have attributes describing the person that registered the request (e.g., "John"), the time of the event (e.g., "30-11-2011:14.55"), the age of the customer (e.g., "45"), and the claimed amount (e.g., "650 euro"). After aligning model and log it is possible to replay the event log on the model. While replaying one can analyze these additional attributes.

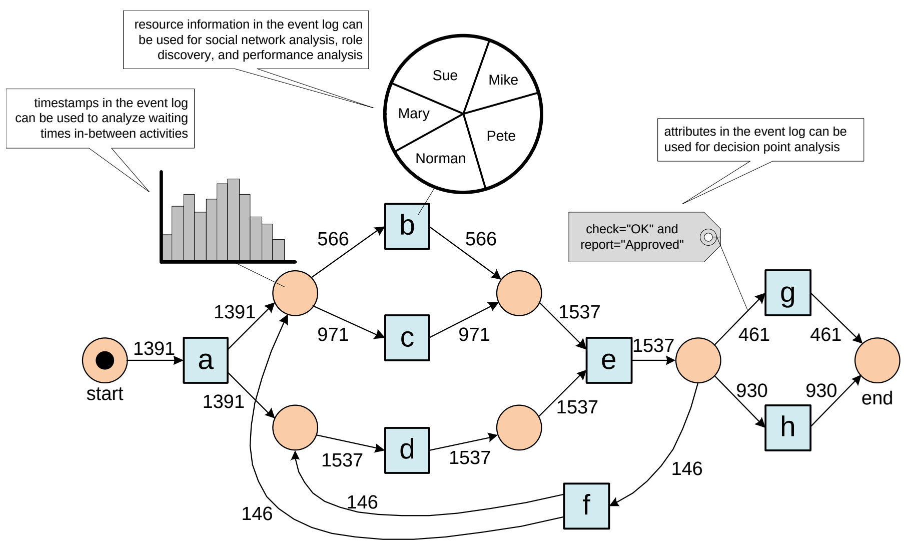

Fig. 3. The process model can be extended using event attributes such as timestamps, resource information, and case data. The model also shows frequencies, e.g., 1537 times a decision was made and 930 cases were rejected.

For example, as Fig. 3 shows, it is possible to analyze waiting times in-between activities. Simply measure the time difference between causally related events and compute basic statistics such as averages, variances, and confidence intervals. This way it is possible to identify the main bottlenecks.

Information about resources can be used to discover roles, i.e., groups of people frequently executing related activities. Here, standard clustering techniques can be used. It is also possible to construct social networks based on the flow of work and analyze resource performance (e.g., the relation between workload and service times).

Standard classification techniques can be used to analyze the decision points in the process model. For example, activity e ("decide") has three possible outcomes ("pay", "reject", and "redo"). Using the data known about the case prior to the decision, we can construct a decision tree explaining the observed behavior.

Figure 3 illustrates that process mining is not limited to control-flow discovery. Moreover, process mining is not restricted to offline analysis and can also be used for predictions and recommendations at runtime. For example, the completion time of a partially handled customer order can be predicted using a discovered process model with timing information.

5. PROCESS MINING CREATES VALUE IN SEVERAL WAYS

After introducing the three types of process mining using a small example, we now focus on the practical value of process mining. As mentioned earlier, process mining is driven by the exponential growth of event data. For example, according to MGI, enterprises stored more than 7 exabytes of new data on disk drives in 2010 while consumers stored more than 6 exabytes of new data on devices such as PCs and notebooks [Manyika et al. 2011].

In the remainder, we will show that process mining can provide value in several ways. To illustrate this we refer to case studies where we used our open-source software package *ProM* [Aalst 2011a]. ProM was created and is maintained by the process mining group at Eindhoven University of Technology. However, research groups from all over the world contributed to it, e.g., University of Padua, Universitat Politècnica de Catalunya, University of Calabria, Humboldt-Universität zu Berlin, Queensland University of Technology, Technical University of Lisbon, Vienna University of Economics and Business, Ulsan National Institute of Science and Technology, K.U. Leuven, Tsinghua University, and University of Innsbruck. Besides ProM there are about 10 commercial software vendors providing process mining software (often embedded in larger tools), e.g., Pallas Athena, Software AG, Futura Process Intelligence, Fluxicon, Businesscape, Iontas/Verint, Fujitsu, and Stereologic.

5.1. Provide Insights

In the last decade, we have applied our process mining software ProM in over 100 organizations. Examples are municipalities (about 20 in total, e.g., Alkmaar, Heusden, and Harderwijk), government agencies (e.g., Rijkswaterstaat, Centraal Justitieel Incasso Bureau, and the Dutch Justice department), insurance related agencies (e.g., UWV), banks (e.g., ING Bank), hospitals (e.g., AMC hospital and Catharina hospital), multinationals (e.g., DSM and Deloitte), high-tech system manufacturers and their customers (e.g., Philips Healthcare, ASML, Ricoh, and Thales), and media companies (e.g., Winkwaves). For each of these organizations, we discovered some of their processes based on the event data they provided. In each discovered process, there were parts that surprised some of the stakeholders. The variability of processes is typically much bigger than expected. Such insights represent a tremendous value as surprising differences often point to waste and mismanagement.

5.2. Improve Performance

As explained earlier, it is possible to replay event logs on discovered or hand-made process models. This can be used for conformance checking and model enhancement. Since most event logs contain timestamps, replay can be used to extend the model with performance information.

Figure 4 illustrates some of the performance-related diagnostics that can be obtained through process mining. The model shown was discovered based on 745 objections against the so-called WOZ ("Waardering Onroerende Zaken") valuation in a Dutch municipality. Dutch municipalities need to estimate the value of houses and apartments. The WOZ value is used as a basis for determining the real-estate property tax. The higher the WOZ value, the more tax the owner needs to pay. Therefore, many citizens appeal against the WOZ valuation and assert that it is too high.

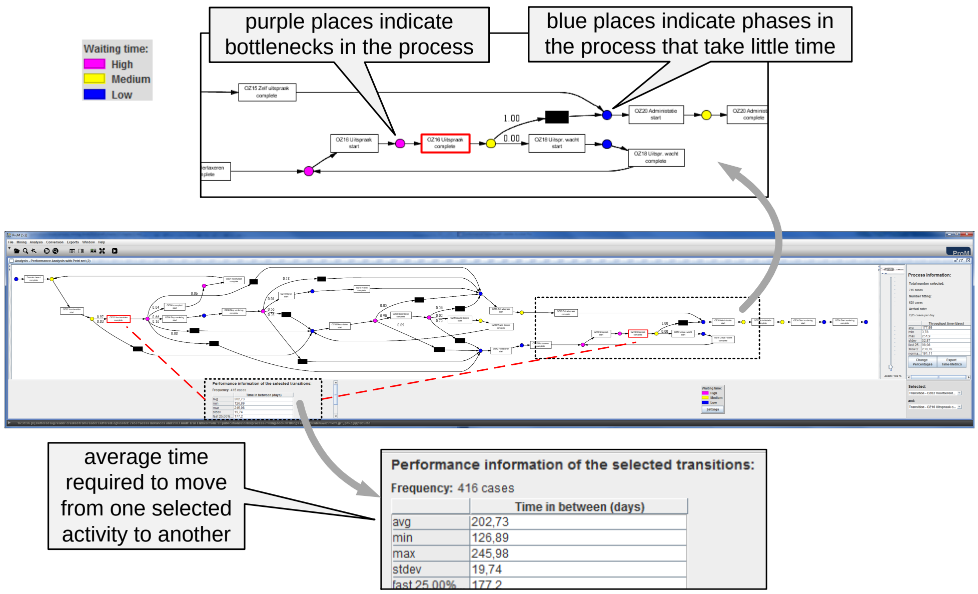

Fig. 4. Performance analysis based on 745 appeals against the WOZ valuation.

Each of the 745 objections corresponds to a process instance. Together these instances generated 9583 events all having timestamps. Figure 4 shows the frequency of the different paths in the model. Moreover, the different stages of the model are colored to show where, on average, most time is spent. The purple stages of the process take most time whereas the blue stages take the least time. It is also possible to select two activities and measure the time that passes in-between these activities. As shown in Fig. 4, on average, 202.73 days pass in-between the completion of activity "OZ02 Voorbereiden" (preparation) and the completion of "OZ16 Uitspraak" (final judgment). This is longer than the average overall flow time which is approx. 178 days. About 416 of the objections (approx. 56%) follow this route; the other cases follow the branch "OZ15 Zelf uitspraak" which, on average, takes less time.

Diagnostics as shown in Fig. 4 can be used to improve processes by removing bottlenecks and rerouting cases. Since the model is connected to event data, it is possible to "drill down" immediately and investigate groups of cases that take more time than others [Aalst 2011a].

5.3. Ensure Conformance

Replay can also be used to check conformance as is illustrated by Fig. 5. Based on 745 appeals against the WOZ valuation, we also compared the normative model and the observed behavior: 628 of the 745 cases can be replayed without encountering any problems. The fitness of the model and log is 0.98876214 indicating that almost all recorded events are explained by the model. Despite the good fitness, ProM clearly shows all deviations. For example, "OZ12 Hertaxeren" (reevaluate property) occurred 23 times while this was not allowed according to the normative model (indicated by the "-23" in Fig. 5). Again it is easy to "drill down" and see what these cases have in common.

The conformance of the appeal process just described is very high (about 99% of events are possible according to the model). We also encountered many processes with

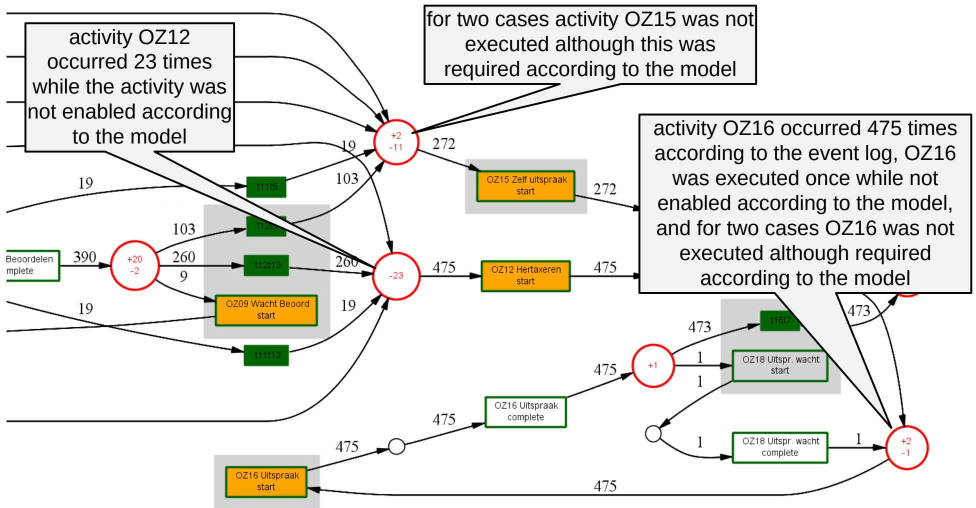

Fig. 5. Conformance analysis showing deviations between event log and process model.

a very low conformance, e.g., it is not uncommon to find processes where only 40% of the events are possible according to the model. For example, process mining revealed that ASML's modeled test process strongly deviated from the real process [Rozinat et al. 2009].

The increased importance of corporate governance, risk and compliance management, and legislation such as the Sarbanes-Oxley Act (SOX) and the Basel II Accord, illustrate the practical relevance of conformance checking. Process mining can help auditors to check whether processes are executed within certain boundaries set by managers, governments, and other stakeholders [Aalst et al. 2010]. Violations discovered through process mining may indicate fraud, malpractice, risks, and inefficiencies. For example, in the municipality where we analyzed the WOZ appeal process, we discovered misconfigurations of their eiStream workflow management system. People also bypassed the system. This was possible because system administrators could manually change the status of cases [Rozinat and Aalst 2008].

5.4. Show Variability

Hand-made process models tend to provide an idealized view on the business process that is modeled. Often such a "PowerPoint reality" has little in common with the real processes that have much more variability. However, to improve conformance and performance, one should not abstract away this variability.

In the context of process mining we often see Spaghetti-like models such as the one shown in Fig. 6. The model was discovered based on an event log containing 24331 events referring to 376 different activities. The event log describes the diagnosis and treatment of 627 gynecological oncology patients in the AMC hospital in Amsterdam. The Spaghetti-like structures are not caused by the discovery algorithm but by the true variability of the process.

Although it is important to confront stakeholders with the reality as shown in Fig. 6, we can also seamlessly simplify Spaghetti-like models. Just like using electronic maps it is possible to seamlessly zoom in and out [Aalst 2011a]. While zooming out, insignificant things are either left out or dynamically clustered into aggregate shapes – like

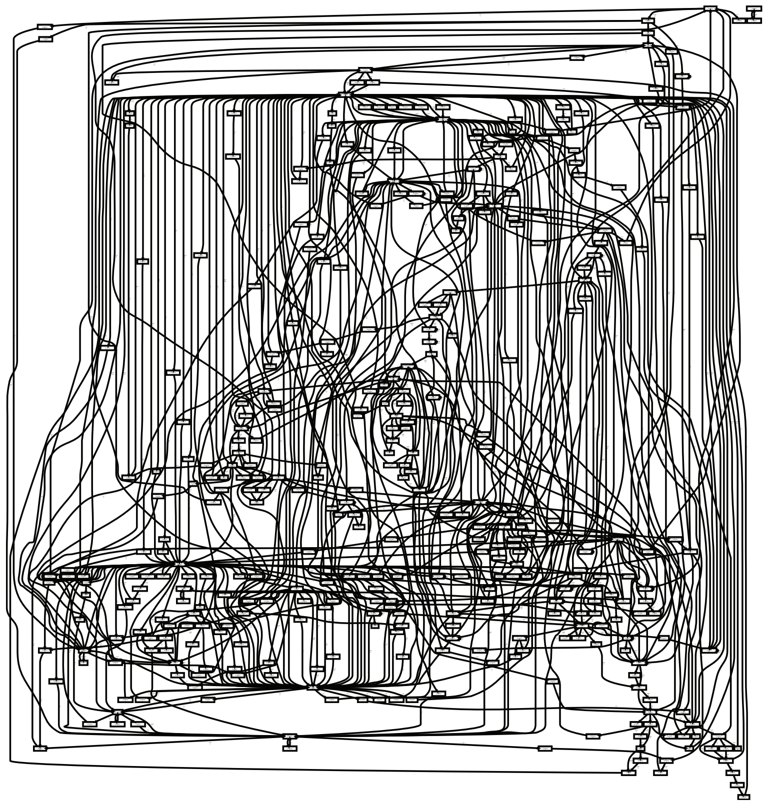

Fig. 6. Process model discovered for a group of 627 gynecological oncology patients.

streets and suburbs amalgamate into cities in Google Maps. The significance level of an activity or connection may be based on frequency, costs, or time.

5.5. Improve Reliability

Process mining can also be used to improve the reliability of systems and processes. For example, since 2007 we have been involved in an ongoing effort to analyze the event logs of the X-ray machines of Philips Healthcare using process mining [Aalst 2011a]. These machines record massive amounts of events. For medical equipment it is essential to prove that the system was tested under realistic circumstances. Therefore, process discovery was used to construct realistic test profiles. Philips Healthcare also used process mining for fault diagnosis. By learning from earlier problems, it is possible to find the root cause for new problems that emerge. For example, using ProM, we have analyzed under which circumstances particular components are replaced. This resulted in a set of signatures. When a malfunctioning X-ray machine exhibits a particular "signature" behavior, the service engineer knows what component to replace.

5.6. Enable Prediction

The combination of historic event data with real-time event data can also be used to predict problems. For instance, Philips Healthcare can anticipate that an X-ray tube in the field is about to fail by discovering patterns in event logs. Hence, the tube can be replaced before the machine starts to malfunction.

Today, many data sources are updated in (near) real-time and sufficient computing power is available to analyze events as they occur. Therefore, process mining is not restricted to off-line analysis and can also be used for online operational support. For a running process instance it is possible to make predictions such as the expected remaining flow time [Aalst 2011a].

6. CONCLUSION

Process mining techniques enable organizations to X-ray their business processes, diagnose problems, and get suggestions for treatment. Process discovery often provides new and surprising insights. These can be used to redesign processes or improve management. Conformance checking can be used to see where processes deviate. This is very relevant as organizations are required to put more emphasis on corporate governance, risks, and compliance. Process mining techniques offer a means to more rigorously check compliance while improving performance.

This article introduced the basic concepts and showed that process mining can provide value in several ways. The reader interested in process mining is referred to the first book on process mining [Aalst 2011a] and the process mining manifesto [TFPM 2012] which is available in 12 languages. Also visit `www.processmining.org` for sample logs, videos, slides, articles, and software.

ACKNOWLEDGMENTS

The author would like to thank the members of the IEEE Task Force on Process Mining and all that contributed to the Process Mining Manifesto and the ProM framework.

REFERENCES

AALST, W. VAN DER 2011a. *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag, Berlin.

AALST, W. VAN DER 2011b. Using Process Mining to Bridge the Gap between BI and BPM. *IEEE Computer 44,* 12, 77–80.

AALST, W. VAN DER, HEE, K. VAN, WERF, J.M. VAN DER, AND VERDONK, M. 2010. Auditing 2.0: Using Process Mining to Support Tomorrow's Auditor. *IEEE Computer 43,* 3, 90–93.

HILBERT, M. AND LOPEZ, P. 2011. The World's Technological Capacity to Store, Communicate, and Compute Information. *Science 332,* 6025, 60–65.

TFPM TASK FORCE ON PROCESS MINING. 2012. Process Mining Manifesto. In *Business Process Management Workshops*, F. Daniel, K. Barkaoui, and S. Dustdar, Eds. Lecture Notes in Business Information Processing Series, vol. 99. Springer-Verlag, Berlin, 169–194.

MANYIKA, J., CHUI, M., BROWN, B., BUGHIN, J., DOBBS, R., ROXBURGH, C., AND BYERS, A. 2011. Big Data: The Next Frontier for Innovation, Competition, and Productivity. McKinsey Global Institute.

MENDLING, J., NEUMANN, G., AND AALST, W. VAN DER 2007. Understanding the Occurrence of Errors in Process Models Based on Metrics. In *Proceedings of the OTM Conference on Cooperative information Systems (CoopIS 2007)*, F. Curbera, F. Leymann, and M. Weske, Eds. Lecture Notes in Computer Science Series, vol. 4803. Springer-Verlag, Berlin, 113–130.

ROZINAT, A. AND AALST, W. VAN DER 2008. Conformance Checking of Processes Based on Monitoring Real Behavior. *Information Systems 33,* 1, 64–95.

ROZINAT, A., DE JONG, I., GÜNTHER, C., AND AALST, W. VAN DER 2009. Process Mining Applied to the Test Process of Wafer Scanners in ASML. *IEEE Transactions on Systems, Man and Cybernetics, Part C 39,* 4, 474–479.