Business Process Simulation Revisited

Wil M.P. van der Aalst

Department of Mathematics and Computer Science,
Eindhoven University of Technology,
P.O. Box 513, NL-5600 MB, The Netherlands
w.m.p.v.d.aalst@tue.nl

Abstract. Computer simulation attempts to "mimic" real-life or hypothetical behavior on a computer to see how processes or systems can be improved and to predict their performance under different circumstances. Simulation has been successfully applied in many disciplines and is considered to be a relevant and highly applicable tool in Business Process Management (BPM). Unfortunately, in reality the use of simulation is limited. Few organizations actively use simulation. Even organizations that purchase simulation software (stand-alone or embedded in some BPM suite), typically fail to use it continuously over an extended period. This keynote paper highlights some of the problems causing the limited adoption of simulation. For example, simulation models tend to oversimplify the modeling of people working part-time on a process. Also simulation studies typically focus on the steady-state behavior of business processes while managers are more interested in short-term results (a "fast forward button" into the future) for operational decision making. This paper will point out innovative simulation approaches leveraging on recent breakthroughs in process mining.

1 Limitations of Traditional Simulation Approaches

Simulation was one of the first applications of computers. The term "Monte Carlo simulation" was first coined in the Manhattan Project during World War II, because of the similarity of statistical simulation to games of chance played in the Monte Carlo Casino. This illustrates that that already in the 1940s people were using computers to simulate processes (in this case to investigate the effects of nuclear explosions). Later Monte Carlo methods were used in all kinds of other domains ranging from finance and telecommunications to games and workflow management. For example, note that the influential and well-known programming language Simula, developed in the 1960s, was designed for simulation. Simulation has become one of the standard analysis techniques used in the context of operation research and operations management. Simulation is particularly attractive since it is versatile, imposes few constraints, and produces results that are relatively easy to interpret. Analytical techniques have other advantages but typically impose additional constraints and are not as easy to use [9]. Therefore, it is no surprise that in the context of *Business Process Management* (BPM), simulation is one of the most established analysis techniques supported by a vast array of tools.

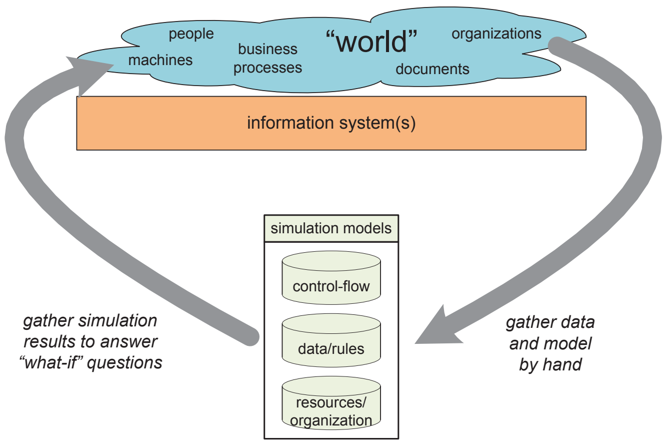

Fig. 1. Classical view on simulation: focus is on steady-state and model is made by hand.

Figure 1 positions *business process simulation* in the context of a "world" supported by information systems. In the "world" consisting of people, organizations, products, processes, machines, etc. *information systems* play an increasingly dominant role. Moreover, there is continuous need for process improvements resulting in a better performance (e.g., better response times, less costs, higher service levels, etc.). Simulation can assist in this. Figure 1 shows the traditional use of simulation were data is gathered and used to parameterize hand-made models. These models are then used for simulation experiments answering "what-if" questions. For simulating business processes at least three *perspectives* need to be modeled: (a) *control-flow*, (b) *data/rules*, and (c) *resource/organization*. The control-flow perspective is concerned with the ordering of activities and uses design artifacts such as sequences, AND/XOR-splits/joins, loops, etc. [1]. The data/rules perspective models decisions made within the process and the role that data plays in these decisions. For simulation it is important not to model the data in too much detail and select the right abstraction level. The resource/organization perspective is concerned with the allocation of activities to resources, availability and speed of resources, and organizational boundaries [21]. In all of this *time* (e.g., the duration of an activity) and *probabilities* (e.g., the likelihood of following a particular path) play an important role. By answering "what-if" questions, managers and users get more insight into the effects of particular decisions.

Although many organizations have tried to use simulation to analyze their business processes at some stage, *few are using simulation in a structured and effective manner*. This may be caused by a lack of training and limitations of existing tools. However, as argued in this paper, there are also several additional and more fundamental problems. First of all, simulation models tend to *over-*

simplify things. In particular the behavior of resources is often modeled in a rather naive manner. People do not work at constant speeds and need to distribute their attention over multiple processes. This can have dramatic effects on the performance of a process [2, 15] and, therefore, such aspects should not be "abstracted away". Second, various *artifacts available are not used as input for simulation*. Modern organizations store events in logs and some may have accurate process models stored in their BPM/WFM systems. Also note that in may organizations, the state of the information system accurately reflects the state of the business processes supported by these systems because of the tight coupling between both. Today such information (i.e., event logs and status data) is rarely used for simulation or a lot of manual work is needed to feed this information into the model. Fortunately, *process mining* can assist in extracting such information and use this to realize performance improvements [4, 7]. Third, the focus of simulation is mainly on "design" while managers would also like to use simulation for *"operational decision making"* (solving the concrete problem at hand rather than some abstract future problem). Fortunately, *short-term simulation* [16, 20, 24] can provide answers for questions related to "here and now". The key idea is to start all simulation runs from the current state and focus the analysis of the transient behavior. This way a "fast forward button" into the future is provided.

In the remainder, we elaborate on the above three problems and discuss some solution approaches grounded in process mining.

2 Oversimplified Simulation Models

Everything should be made as simple as possible,
but not one bit simpler.
Albert Einstein (1879-1955)

Simulation can be used to predict the performance under various circumstances, e.g., different business process re-engineering alternatives can be compared with the current situation. The value of such predictions stands or falls with the quality of the simulation model. Unfortunately, in many situations the quality of the simulation model leaves much to be desired. Basically, there are three problems: (a) the process is modeled incorrectly, (b) not enough data was collected to be able to parameterize the model, and (c) the language does not allow for the modeling of more subtle behaviors. The first two problems can be addressed by training people and a better validation of the model, e.g., by comparing the simulation results with real data. Here process mining can help as will be discussed in later sections. In this section, we focus on the last problem.

Probably the biggest problem of current business simulation approaches is that *human resources are modeled in a very naive manner*. As a result, it is not uncommon that the simulated model predicts flow times of minutes or hours while in reality flow times are weeks or even months. Therefore, we list some of the main problems encountered when modeling resources in current simulation tools. These problems stem from the fact that resources cannot be modeled adequately.

People are involved in multiple processes. In practice there are few people that only perform activities for a single process. Often people are involved in many different processes, e.g., a manager, doctor, or specialist may perform tasks in a wide range of processes. However, simulation often focuses on a single process. Suppose a manager is involved in 10 different processes and spends about 20 percent of his time on the process that we want to analyze. In most simulation tools it is impossible to model that a resource is only available 20 percent of the time. Hence, one needs to assume that the manager is there all the time and has a very low utilization. As a result the simulation results are too optimistic. In the more advanced simulation tools, one can indicate that resources are there at certain times in the week (e.g., only on Monday). This is also an incorrect abstraction as the manager distributes his work over the various processes based on priorities and workload. Suppose that there are 5 managers all working 20 percent of their time on the process of interest. One could think that these 5 managers could be replaced by a single manager (5*20%=1*100%). However, from a simulation point of view this is an incorrect abstraction. There may be times that all 5 managers are available and there may be times that none of them are available.

People do not work at a constant speed. Another problem is that people work at different speeds based on their workload, i.e., it is not just the distribution of attention over various processes, but also their absolute working speed that determines their capacity for a particular process. There are various studies that suggest a relation between workload and performance of people. A well-known example is the so-called Yerkes-Dodson law [23]. The Yerkes-Dodson law models the relationship between arousal and performance as a $\cap$-shaped curve. This implies that for a given individual and a given type of tasks, there exists an optimal arousal level. This is the level where the performance has its maximal value. Thus work pressure is productive, up to a certain point, beyond which performance collapses. Although this phenomenon can be easily observed in daily life, today's business process simulation tools do not support the modeling of workload dependent processing times.

People tend to work part-time and in batches. As indicated earlier, people may be involved in different processes. Moreover, they may work part-time (e.g., only in the morning). In addition to their limited availabilities, people have a tendency to work in batches (cf. Resource Pattern 38: Piled Execution [21]). In any operational process, the same task typically needs to be executed for many different cases (process instances). Often people prefer to let work-items related to the same task accumulate, and then process all of these in one batch. In most simulation tools a resource is either available or not, i.e., it is assumed that a resource is eagerly waiting for work and immediately reacts to any work-item that arrives. Clearly, this does not do justice to the way people work in reality. For example, consider how and when people reply to e-mails. Some people handle e-mails one-by-one when they arrive while others process their e-mail at fixed times in batch. Related is the fact that calendars and shifts are typically ignored in simulation tools. While holidays, lunch breaks, etc. can heavily impact the performance of a process, they are typically not incorporated in the simulation model.

Priorities are difficult to model. As indicated above, people are involved in multiple processes and even within a single process different activities and cases may compete for resources. One process may be more important than another and get priority. Another phenomenon is that in some processes cases that are delayed get priority while in other processes late cases are “sacrificed” to finish other cases in time. People need to continuously choose between work-items and set priorities. Although important, this is typically not captured by simulation models.

Process may change depending on context. Another problem is that most simulation tools assume a stable process and organization and that neither of them change over time. If the flow times become too long and work is accumulating, resources may decide to skip certain activities or additional resources may be mobilized. Depending on the context, processes may be configured differently and resources may be deployed differently. In [5] it is shown that such “second order dynamics” heavily influence performance.

The problems stem from oversimplified models. Note that although more than 40 resource patterns have been identified to describe the functionality of resource allocation mechanisms in the context of workflow management systems [21], few of these patterns are supported by today’s business process simulation tools.

3 Learning from Event Logs

Learning is not compulsory ... neither is survival.
William Edwards Deming (1900-1993)

As discussed in the previous section, simulation models tend not to capture certain aspects or stick to an idealized variant of the real process. This can be partly addressed by better modeling techniques, e.g., additional parameters describing the resource characteristics. However, to adequately set these parameters and to make sure that processes are modeled accurately, we propose to also *exploit the information available in event logs.*

More and more information about (business) processes is recorded by information systems in the form of so-called “event logs” (e.g., transaction logs, audit trails, databases, message logs). As mentioned earlier, IT systems are becoming more and more intertwined with the processes they support, resulting in an "explosion" of available data that can be used for analysis purposes.

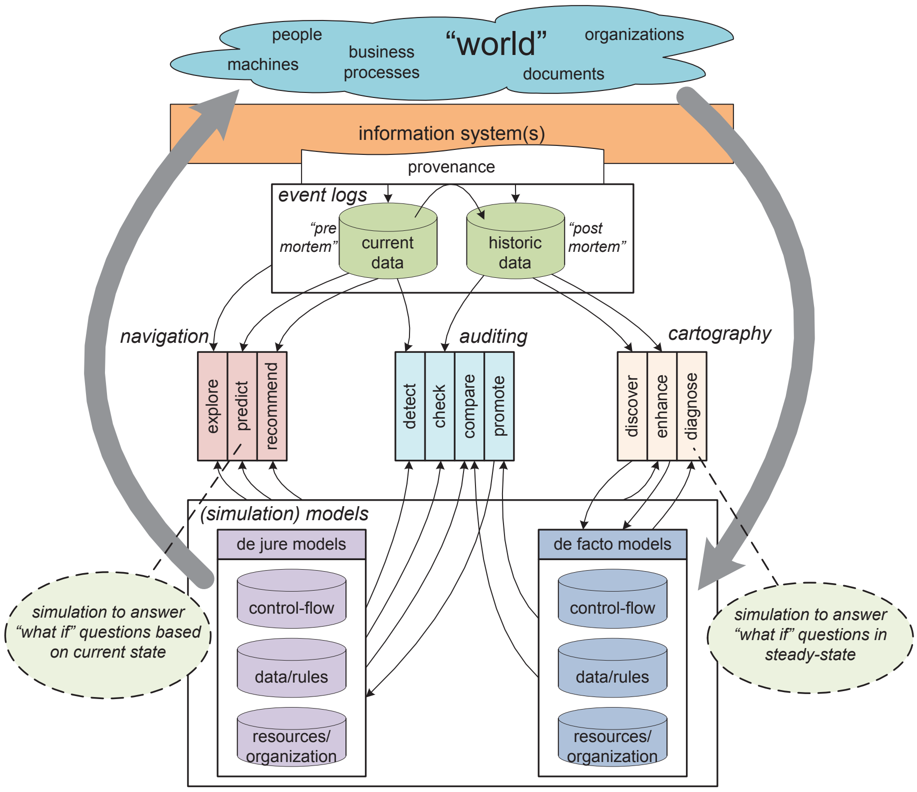

Fig. 2. Advanced business process simulation put into the context of process mining.

To illustrate the role that event logs can play, let us first explain Figure 2. We assume the existence of a collection of information systems that are supporting a "world" composed of business processes, people, organizations, etc. The *event data* extracted from such systems are the starting point for *process mining*. Note that Figure 2 distinguishes between *current data* and *historic data*. The former refers to events of cases (i.e., process instances) that are still actively worked on ("pre mortem"). The latter refers to events of completed cases, i.e., process instances that cannot be influenced anymore ("post mortem"). The historic data ("post mortem") can be any collection of events where each event refers to an instance (i.e., case), has a name (e.g., activity name), and has a timestamp. Note that some process mining techniques abstract from time. However, in the context of business process simulation these timestamps are of the utmost importance.

The current data ("pre mortem") can be used to construct a well defined starting point for simulation. This is of particular importance for predictions in the near future.

The collection of event data is becoming more important. One the one hand, more and more event data are available. On the other hand, organizations depend on such data; not only for performance measurement, but also for auditing. We use the term *business process provenance* [10, 11] to refer to the systematic collection of the information needed to reconstruct what has actually happened. The term signifies that for most organizations it is vital that "history cannot be rewritten or obscured". From an auditing point of view the systematic, reliable, and trustworthy recording of events is essential. Therefore, we propose to collect (whenever possible) provenance data outside of the operational information system(s) as shown in Figure 2. This means that events need to be collected and stored persistently. Note that semantics play an important role here, i.e., *events need to refer to a commonly agreed-upon ontology* [14].

The lower part of Figure 2 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 the actual reality with all of its intricacies (policy violations, inefficiencies, fraud, etc.). Both types of models may cover one or more perspectives and thus describe control-flow, time, data, organization, resource, and/or cost aspects. For process mining one can focus on a particular perspective. However, when the goal is to build simulation models all factors influencing performance need to be taken into account (e.g., when measuring utilization and response times, it is not possible to abstract from resources and focus on control-flow only). Simulation models can be based on a mixture of "de jure" and "de facto" information. The key idea of process mining is to not simply rely on de jure models that may have little to do with reality. Therefore, the goal is to shift more to "de facto models for simulation"; this will save time and increase quality.

In Figure 2 three main categories of activities have been identified: *cartography*, *auditing*, and *navigation*. The individual activities are briefly described below.

1. **Discover.** The discovery of good process models from events logs - comparable to geographic maps - remains challenging. Process discovery techniques can be used to discover process models (e.g., Petri nets) from event logs [4, 7].
2. **Enhance.** Existing process models (either discovered or hand-made) need to be related to events logs such that these models can be enhanced by making them more faithful or by adding new perspectives based on event data. By combining historic data and pre-existing models, these models can be repaired (e.g., a path that is never taken is removed) or extended (e.g., adding time information extracted from logs).
3. **Diagnose.** Models (either de jure or de facto) need to be analyzed using existing model-based analysis techniques, e.g., process models can be checked

for the absence of deadlocks or simulated to estimate cycle times. Probably the most widely used model-based analysis technique is simulation.

4. **Detect.** For on-line auditing, de jure models need to be compared with current data (events of running process instances) and deviations of such partial cases should to be detected at runtime. By replaying the observed events on a model, it is possible to do conformance checking while the process is unfolding.
5. **Check.** Similarly, historic "post mortem" data can be cross-checked with de jure models. For this conformance checking techniques are used that can pinpoint deviations and quantify the level of compliance [18].
6. **Compare.** De facto models can be compared with de jure models to see in what way reality deviates from what was planned or expected.
7. **Promote.** Based on an analysis of the differences between a de facto model and a de jure model, it is possible to promote parts of the de facto model to a new de jure model. By promoting proven "best practises" to the de jure model, existing processes can be improved. For example, a simulation model may be improved and calibrated based on elements of a de facto model.
8. **Explore.** The combination of event data and models can be used to explore business processes. Here new forms of interactive process visualization can be used (visual analytics).
9. **Predict.** By combining information about running cases with models (discovered or hand-made), it is possible to make predictions about the future, e.g., the remaining flow time and the probability of success. Here simulation plays an important role. This will be elaborated in Section 4.
10. **Recommend.** The information used for predicting the future can also be used to recommend suitable actions (e.g. to minimize costs or time). The goal is to enable functionality similar to the guidance given by navigation systems like TomTom, but now in the context of BPM.

The first three activities are grouped under the term "cartography". Over time cartographers have improved their skills and techniques to create maps thereby addressing problems such as clearly representing desired traits, eliminating irrelevant details, reducing complexity, and improving understandability. Today, geographic maps are digital and of high quality. People can seamlessly zoom in and out using the interactive maps (cf. navigation systems like TomTom and services linked to Google Maps). Moreover, all kinds of information can be projected on these interactive maps (e.g., traffic jams, etc.). Process models can be seen as the "maps" describing the operational processes of organizations. Process mining techniques can be used to generate such maps. These maps can be simple and without executable semantics. However, as shown in [19] also simulation models can be discovered.

The next four activities are grouped under the term "auditing" as they compare normative/modeled behavior with real/recorded behavior. This does not involve simulation; however, these activities may help to increase the quality of discovered/hand-made simulation models.

The last three activities are grouped under the term "navigation". Navigation systems have proven to be quite useful for many drivers. People increasingly rely on the devices of TomTom, Garmin and other vendors and find it useful to *get directions* to go from A to B, know the *expected arrival time*, learn about *traffic jams* on the planned route, and be able to *view maps* that can be *customized* in various ways (zoom-in/zoom-out, show fuel stations, speed limits, etc.). However, when looking at business processes and their information systems, *such information is typically lacking*. Fortunately, a combination of process mining and simulation can help to provide navigation capabilities. The next section focuses on this.

4 Operational Support

If you don't know where you are going, any road will get you there.
Lewis Carroll (1832-1898)

Figure 2 illustrated that event logs can be used for all kinds of analysis, e.g., event logs can be used to discover and improve simulation models. In this section, we focus on *short-term simulation*, i.e., a detailed analysis of the near future based on the current state. Traditionally, business process simulation is mainly used for steady-state analysis and not for operational decision making. To explain the importance of short-term simulation, we first elaborate on the difference between *transient analysis* and *steady-state analysis*.

The key idea of simulation is to execute a model repeatedly. The reason for doing the experiments repeatedly, is to not come up with just a single value (e.g., "the average response time is 10.36 minutes") but to provide confidence intervals (e.g., "the average response time is with 90 percent certainty between 10 and 11 minutes"). For transient analysis the focus is on the initial part of future behavior, i.e., starting from the initial state the "near future" is explored. For transient analysis the initial state is very important. If the simulation starts in a state with long queues of work, then in the near future flow times will be long and it may take some time to get rid of the backlog. For steady-state analysis the initial state is irrelevant. Typically, the simulation is started "empty" (i.e., without any cases in progress) and only when the system is filled with cases the measurements start.

Steady-state analysis is most relevant for answering strategic and tactical questions. Transient analysis is most relevant for operational questions. Lion's share of contemporary simulation support aims at steady-state analysis and hence at strategic and tactical decision making. We advocate *more emphasis on simulation for operational decision making*. Therefore, we elaborate on short-term simulation and relate this to process mining and operational support.

Figure 3 shows the input used for operational support. *Historic data*, i.e., event logs, can be used to discover new models and to enhance existing models. This was already discussed in the previous section. The learned models can be combined with *current data* (i.e., states of cases and partial execution traces) to *detect* deviations, *predict* performance, and to *recommend* decisions. Predictions may be based on regression models [12]. However, to predict more complex

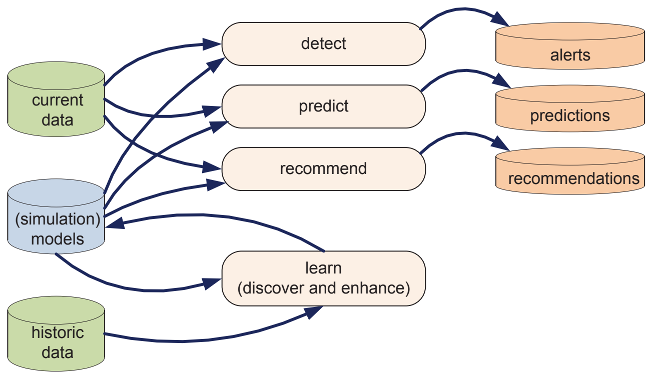

Fig. 3. Overview of operational support and the different types of data used.

dynamic behavior, simulation can be used. In this paper, we distinguish between operational support at the *instance level* and at the *aggregate level*. The instance level focuses on a single case, e.g., a particular loan application that is being processed. It may be detected that the application is delayed and because of this an alert is generated. Moreover, for the partially executed loan application it may be predicted that the expected remaining processing time is two weeks and that therefore it is recommended to bypass an external credit check. Unlike recommendations and predictions at the instance level, operational support at the aggregate level is concerned with the whole process (or even a set of processes). Problems are now detected at the aggregate level ("response times are too long"). Moreover, predictions and recommendations are at the process level and do not refer to particular instances.

Table 1 provides examples of operational support questions. Both levels (instance level and aggregate level) are discussed in the remainder.

Operational support at the instance level. Figure 4 illustrates the three types of operational support. Starting point is some model and a partial trace. Note that the model is typically learned using classical process mining techniques. The partial trace refers to a case that is running. The left-hand side of Figure 4 shows a partial trace $\langle A, B \rangle$. Although Figure 4 does not show timestamps, resources, data, etc., these may be relevant for operational support.

For the case shown in Figure 4, we know that A and B occurred, but we do not know its future. Suppose now that the partial trace $\langle A, B \rangle$ is not possible according to the model. In this case, the operational support system should generate an alert. Another possibility would be that B took place three weeks after A while this should happen within one week. In such a case another notification could be sent to the responsible case manager. Such scenarios correspond to the *check* activity mentioned before. Figure 4 also illustrates the goal of *predictions*.

Table 1. Examples of various types of operational support at the instance level and the aggregate level.

type of operational support	*instance level*	*aggregate level*
detect	Partially executed cases are monitored. As soon as a deviation occurs (e.g., a task is skipped or too late) an alert is given.	Processes are monitored as a whole and as soon as a deviation occurs (e.g., the average response times are too high or too many cases are in the pipeline) an alert is given.
predict	Predictions are made for specific cases, e.g., after each step the expected remaining processing time of the case is given. Predictions may also refer to costs and quality, e.g., the likelihood of succes for a particular instance. Short-term simulation can be used to generate such instance-level predictions.	Predictions are made for one process or a collection of processes. For example, it is predicted what the average flow time will be in the next two weeks. Predictions at the aggregate level may also refer to utilization ("How busy will people be next week?"), costs ("Will we reach the break-even point in this quarter?"), service levels, etc.
recommend	Predictions at the instance level can be turned into recommendations by exploring the effect of various decisions. For example, different routing choices can be simulated to predict the effect of such choices. Similarly, the effect of various allocation choices can be compared using simulation.	Predictions at the aggregate level can be used to generate recommendations. The effect of each decision can be analyzed using short-term simulation. For example, it may be recommended to temporarily hire two additional workers to avoid excessive waiting times.

Given the current state of a case, the model is used to make some kind of prediction [3, 6]. For example, given the $\langle A, B \rangle$ trace it could be predicted that the remaining processing time is ten days. This prediction would be based on historic information both in the partial trace and in the event log used to learn the model. For the actual prediction a simple regression model can be used. However, for more complex scenarios, *short-term simulation* is a more likely option. Predictions are not restricted to time, but can also refer to costs, probability of a particular outcome, resource availability, etc. Closely related to predictions are *recommendations* [3, 22]. The main difference is that recommendations suggest the next action based on possible continuations of the case. Based on the model, one can try all possible actions and see which one would lead to the best (predicted) performance. Note that recommendations are not only used for determining the next task, but also for allocating resources to work-items or for timing a particular action.

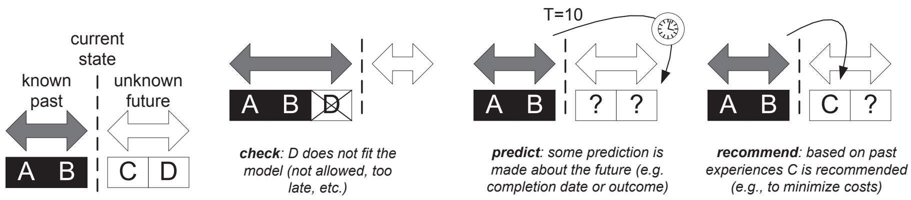

Fig. 4. Operational support at the instance level [3].

Operational support at the aggregate level. In Figure 4 analysis is done at the instance level. However, many operational decisions transcend the level of an individual case. Decisions like temporarily adding two workers or stimulate overwork are made at the level of one or more processes rather than a single case. Short-term simulation is particularly useful for predictions at the aggregate level. Here, simple regression models are unable to capture queueing effects, dependencies, and typical work patterns.

Short-term simulation starts from the current state [16, 20, 24]. When a process-aware information system is present, it is relatively easy to extract the current state from the system and to upload this into the simulation model. By modifying the simulation model, various "what-if" scenarios can be investigated. For example, one can add or remove resources, skip activities, etc. and see what the effect is. Because the simulation experiments for these scenarios start from the current state of the actual system, they provide a kind of "fast-forward button" showing what will happen in the near future, to support operational decision making. For instance, based on the predicted system behavior, a manager may decide to hire more personnel or stop accepting new cases.

5 Conclusion and Further Reading

The goal of this keynote paper is to provide a *critical analysis of the mainstream simulation approaches* for process management. On the one hand, the paper is based on practical experiences in numerous simulation projects (cf. [17] for examples). These experiences showed amongst others that it is almost impossible to adequately model resources in contemporary simulation tools. On the other hand, various process mining projects showed that reality rarely matches the expectations of the modeler. Models tend to describe idealized/unrealistic views on the business processes at hand. These practical experiences with simulation and process mining resulted in a better understanding of the pitfalls of traditional business process analysis. Some of the lessons learned have been reported. Moreover, as shown, business process simulation can benefit from *recent breakthroughs in process mining*.

Several of the ideas presented in this paper have been realized in the context of *ProM* (www.processmining.org, [8]) and *YAWL* (www.yawlfoundation.org, [13]). To conclude this paper, we provide pointers to papers detailing these results.

In [3] a concrete approach to operational support is given. This has been implemented in ProM and time-based predictions and recommendations are given by learning a transition system annotated with time information [6]. The focus in [3] is restricted to individual cases and temporal aspects.

In [15] it is shown how event logs can be used to learn about the behavior of people. For example, through process mining one can find empirical evidence for the Yerkes-Dodson law [23] and parameterize the corresponding simulation models.

ProM provides comprehensive support for the automated discovery of simulation models based on event logs. In [19] it is shown how different perspectives can be discovered and merged into one overall simulation model.

While the focus in [19] is on simulation models for steady-state analysis, the focus of [20] is on short-term simulation, i.e., transient analysis. This is achieved by an integration of ProM and YAWL. The workflow model, event log, and current state information provided by the workflow system YAWL are used by ProM to generate simulation models. These models are simulated using CPN Tools. Key element is that the simulation model is called continuously while using the latest state information. This way a "fast-forward button" is added to YAWL that allows users and manager explore the near future.

One of the key problems when using business process simulation is the fact that it is unrealistic to assume that people are continuous available. Availability and work-speed are fluid. As shown in [2], it is important to capture and parameterize this "fluidity" as it has a dramatic effect on flow times, etc.

The papers mentioned above present innovations in business process simulation. Although quite some work has been done in the context of ProM and YAWL, it remains crucial to further improve techniques and tools to better capture faithful simulation models. Hopefully this will stimulate more organizations to reap the benefits of business process simulation.

Acknowledgments. The author would like to thank all the people that contributed to the development of ProM and YAWL. This paper refers to simulation techniques developed together with Joyce Nakatumba, Anne Rozinat, Moe Wynn, Ronny Mans, Minseok Song, and several others.

References

1. W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros. Workflow Patterns. *Distributed and Parallel Databases*, 14(1):5–51, 2003.
2. W.M.P. van der Aalst, J. Nakatumba, A. Rozinat, and N. Russell. Business Process Simulation: How to get it right? In J. vom Brocke and M. Rosemann, editors, *Handbook on Business Process Management*, International Handbooks on Information Systems, pages 317–342. Springer-Verlag, Berlin, 2010.
3. W.M.P. van der Aalst, M. Pesic, and M. Song. Beyond Process Mining: From the Past to Present and Future. In B. Pernici, editor, *Advanced Information Systems Engineering, Proceedings of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10)*, volume 6051 of *Lecture Notes in Computer Science*, pages 38–52. Springer-Verlag, Berlin, 2010.
4. 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.
5. W.M.P. van der Aalst, M. Rosemann, and M. Dumas. Deadline-based Escalation in Process-Aware Information Systems. *Decision Support Systems*, 43(2):492–511, 2007.
6. 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.
7. W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster. Workflow Mining: Discovering Process Models from Event Logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9):1128–1142, 2004.
8. W.M.P. van der Aalst and. B. van Dongen, C.W. Günther, A. Rozinat, E. Verbeek, and T. Weijters. ProM: The Process Mining Toolkit. In A.K.A. de Medeiros and B. Weber, editors, *Business Process Management Demonstration Track (BPMDemos 2009)*, volume 489 of *CEUR Workshop Proceedings*, pages 1–4. CEUR-WS.org, 2009.
9. J.A. Buzacott. Commonalities in Reengineerd Business Processes: Models and Issues. *Management Science*, 42(5):768–782, 1996.
10. F. Curbera, Y. Doganata, A. Martens, N. Mukhi, and A. Slominski. Business Provenance: A Technology to Increase Traceability of End-to-End Operations. In R. Meersman and Z. Tari, editors, *Proceedings of the 16th International Conference on Cooperative Information Systems, CoopIS 2008, OTM 2008, Part I*, volume 5331 of *Lecture Notes in Computer Science*, pages 100–119. Springer-Verlag, Berlin, 2008.
11. S. Davidson, S. Cohen-Boulakia, A. Eyal, B. Ludaescher, T. McPhillips, S. Bowers, M. Anand, and J. Freire. Provenance in Scientific Workflow Systems. *Data Engineering Bulletin*, 30(4):44–50, 2007.
12. B.F. van Dongen, R.A. Crooy, and W.M.P. van der Aalst. Cycle Time Prediction: When Will This Case Finally Be Finished? In R. Meersman and Z. Tari, editors, *Proceedings of the 16th International Conference on Cooperative Information Systems, CoopIS 2008, OTM 2008, Part I*, volume 5331 of *Lecture Notes in Computer Science*, pages 319–336. Springer-Verlag, Berlin, 2008.

13. A.H.M. ter Hofstede, W.M.P. van der Aalst, M. Adams, and N. Russell. *Modern Business Process Automation: YAWL and its Support Environment*. Springer-Verlag, Berlin, 2010.
14. A.K. Alves de Medeiros and W.M.P. van der Aalst. Process Mining towards Semantics. In T.S. Dillon, editor, *Advances in Web Semantics I*, volume 4891 of *Lecture Notes in Computer Science*, pages 35–80. Springer-Verlag, Berlin, 2008.
15. J. Nakatumba and W.M.P. van der Aalst. Analyzing Resource Behavior Using Process Mining. In S. Rinderle-Ma, S. Sadiq, and F. Leymann, editors, *BPM 2009 Workshops, Proceedings of the Fifth Workshop on Business Process Intelligence (BPI'09)*, volume 43 of *Lecture Notes in Business Information Processing*, pages 69–80. Springer-Verlag, Berlin, 2010.
16. H.A. Reijers and W.M.P. van der Aalst. Short-Term Simulation: Bridging the Gap between Operational Control and Strategic Decision Making. In M.H. Hamza, editor, *Proceedings of the IASTED International Conference on Modelling and Simulation*, pages 417–421. IASTED/Acta Press, Anaheim, USA, 1999.
17. H.A. Reijers and W.M.P. van der Aalst. The Effectiveness of Workflow Management Systems: Predictions and Lessons Learned. *International Journal of Information Management*, 25(5):458–472, 2005.
18. 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.
19. A. Rozinat, R.S. Mans, M. Song, and W.M.P. van der Aalst. Discovering Simulation Models. *Information Systems*, 34(3):305–327, 2009.
20. A. Rozinat, M. Wynn, W.M.P. van der Aalst, A.H.M. ter Hofstede, and C. Fidge. Workflow Simulation for Operational Decision Support. *Data and Knowledge Engineering*, 68(9):834–850, 2009.
21. N. Russell, W.M.P.van der Aalst, A.H.M. ter Hofstede, and D. Edmond. Workflow Resource Patterns: Identification, Representation and Tool Support. In O. Pastor and J. Falcao e Cunha, editors, *Proceedings of the 17th Conference on Advanced Information Systems Engineering (CAiSE'05)*, volume 3520 of *Lecture Notes in Computer Science*, pages 216–232. Springer-Verlag, Berlin, 2005.
22. H. Schonenberg, B. Weber, B.F. van Dongen, and W.M.P. van der Aalst. Supporting Flexible Processes Through Recommendations Based on History. In M. Dumas, M. Reichert, and M.C. Shan, editors, *International Conference on Business Process Management (BPM 2008)*, volume 5240 of *Lecture Notes in Computer Science*, pages 51–66. Springer-Verlag, Berlin, 2008.
23. C.D. Wickens. *Engineering Psychology and Human Performance*. Harper, 1992.
24. M. Wynn, A. Rozinat, W.M.P. van der Aalst, A.H.M. ter Hofstede, and C. Fidge. Chapter 17: Process Mining and Simulation. In *Modern Business Process Automation: YAWL and its Support Environment*, pages 437–457. Springer-Verlag, Berlin, 2010.