Liquid Business Process Model Collections

Wil M.P. van der Aalst[1,2], Marcello La Rosa[2,3], Arthur H.M. ter Hofstede[2,1], Moe T. Wynn[2]

[1] Eindhoven University of Technology, Eindhoven, The Netherlands
w.m.p.v.d.aalst@tue.nl
[2] Queensland University of Technology, Brisbane, Australia
{m.larosa, a.terhofstede, m.wynn}@qut.edu.au
[3] NICTA Queensland Research Lab, Brisbane, Australia

Abstract. Many organizations realize that increasing amounts of data ("Big Data") need to be dealt with intelligently in order to compete with other organizations in terms of efficiency, speed and services. The goal is not to collect as much data as possible, but to turn event data into valuable insights that can be used to improve business processes. However, data-oriented analysis approaches fail to relate event data to process models. At the same time, large organizations are generating piles of process models that are disconnected from the real processes and information systems. These models are often just used for documentation and discussion. Even in more mature organizations using analysis techniques like simulation, models are often not connected to event data. In this chapter we propose to manage large collections of process models and event data in an integrated manner. Observed and modeled behavior need to be continuously compared and aligned. This results in a "liquid" business process model collection, i.e. a collection of process models that is in sync with the actual organizational behavior. The collection should self-adapt to evolving organizational behavior and incorporate relevant execution data (e.g.

process performance and resource utilization) extracted from the logs, thereby allowing insightful reports to be produced from factual organizational data.

Keywords: business process management, process mining, process model collection, process model.

Introduction

Business Process Management (BPM) has become an established discipline (Dumas et al. 2013, van der Aalst 2013a) dedicated to the way an organization identifies, captures, analyses, improves, implements and monitors its business processes. Through the management of the *process lifecycle*, BPM influences the effectiveness and efficiency of a corporation and is a significant contributor to its overall performance and competitiveness. Business processes are thus seen as strategic corporate assets and, in the case of comprehensive supply chains, complex call center operations or advanced distribution networks, can represent *multi-million dollar assets* (Gotts 2010). As processes determine how an organization operates, what activities need to be fulfilled and what data and resources are required for their successful execution, they are crucial to a plethora of key performance indicators. This importance of processes has motivated organizations to *significantly invest* in methods, tools and techniques facilitating process lifecycle management. For example, Wolf and Harmon (2012) report that 37% of organizations surveyed spend more than USD 500,000, and 4% more than USD 10 million, on investments in business process analysis, management, monitoring, redesign and improvement, and similar amounts for related software acquisitions.

At the core of the process lifecycle are conceptual models of processes that help involved

stakeholders gain a shared understanding of their processes. Such visual depictions are called *process models*. A process model is a directed graph describing the triggers, activities, data and resources of a business process. Such models can be used for documentation, discussion, enactment (e.g., to configure a BPM system), and various types of analysis. *Simulation* is a classical model-based analysis technique used for "what-if" analysis. Given an "as-is" model describing the current situation, multiple "to-be" models can be constructed to explore different alternatives while measuring Key Performance Indicators (KPIs) like flow times, utilization, response times, faults, risks, and costs. The value of such analyses stands or falls on the quality of the "as-is" model: Is the simulation model able to capture reality?

As the ultimate source of process information, a process model *informs critical decisions* such as investments related to process-aware information systems, complex organizational re-designs or crucial compliance assessments (Davies et al. 2006). These requirements demand that process models accurately reflect the corresponding real-world processes, i.e. the actual *organizational behavior*.

As a consequence, a substantial research branch of BPM, namely *process mining* (van der Aalst 2011, IEEE Task Force on Process Mining 2012), has been dedicated to techniques for extracting organizational behavior from *event logs* and using this information to enhance existing process models (Fahland and van der Aalst 2012) or discover new ones (van Dongen et al. 2009). Event logs are recorded by a variety of IT systems commonly available within organizations, such as Enterprise Resource Planning (ERP) systems, Content Management Systems (CMSs), Customer Relationship Management (CRM) systems, Database Management Systems (DBMSs) or e-mail servers. Process mining is fueled by the incredible growth of event data (Hilbert and Lopez 2011).

In the practical deployment of process modeling, however, a new challenge emerges, i.e. scaling up or *process modeling in the large* (Raduescu et al. 2006, Rosemann 2006, van der Aalst 2013a). For large organizations it is common to maintain collections of thousands of complex process models which all need to be managed appropriately to cater for the various demands of their stakeholders. For instance, Suncorp, the largest Australian insurer, manages a collection of 3,000+ process models with models ranging from 25 to 500 activities (La Rosa et al. 2013).

These large organizations employ some form of process model repository management, e.g. Suncorp uses ARIS. However, such forms of repository management are often not adequate as the models in these repositories tend:

Issue 1: to be based on *subjective* perceptions about the real processes rather than actual *objective* process data,

Issue 2: to be *out of sync* with organizational behavior, as the frequency of real world process changes is such that cost-effective process model updates are not possible,

Issue 3: to be designed for the most *common purposes* of the stakeholders instead of catering for specific demands of individual stakeholders.

These three issues lead to *severe limitations* of process model collections, dramatically compromising the quality of business decisions that are made based on these artifacts. Current process mining techniques are not adequate to address these issues, since they focus on *single* process models as opposed to collections thereof. Concomitantly, recently-emerged research techniques for managing large process model collections (Dijkman et al. 2012), are concerned with challenges such as how to identify similar models in a repository (Dijkman et al. 2011), merge these models (La Rosa et al. 2013) and modularize them (Reijers et al. 2011), but have never considered organizational behavior despite the wide availability of event logs in today's

organizations.

This chapter proposes the new notion of a *"liquid" business process model/log collection*, i.e. a collection of process models that:

i) is *aligned* with the organizational behavior, as recorded in event logs,

ii) can *self-adapt* to evolving organizational behavior, thereby consistently remaining current and relevant,

iii) incorporates relevant *execution data* (e.g. process performance and resource allocations) extracted from the logs, thereby allowing insightful reports to be produced from factual organizational data.

Such collections provide a rich source of input for various types of analysis, including process mining and simulation.

The remainder of the chapter is organized as follows. Section 2 provides an overview of BPM by discussing selected BPM use cases. The process mining spectrum and the importance of aligning observed and modeled behavior are discussed in Section 3 whereas the management techniques for process model collections are discussed in Section 4. Section 5 elaborates on the main innovations needed to make business process model collections "liquid" using event data. Section 6 discusses the different challenges in terms of five research streams. Initial tool support realized through ProM and Apromore is described in Section 7. Section 8 concludes the chapter.

BPM Use Cases

In this chapter we propose a "liquid" business process model collection where modeled and observed behaviors are aligned and multiple evolving processes are considered. To position such liquid business process model and event data collections, we first provide an overview of

classical BPM approaches using typical use cases.

In (van der Aalst 2013a) twenty use cases are used to structure the BPM discipline and to show "how, where, and when" BPM techniques can be used. These are summarized in Figure 1. Models are depicted as pentagons marked with the letter **M**. A model may be descriptive (**D**), normative (**N**), and/or executable (**E**). A "**D|N|E**" tag inside a pentagon means that the corresponding model is descriptive, normative, or executable. Configurable models are depicted as pentagons marked with **CM**. Event data (e.g., an event log) are denoted by a disk symbol (cylinder shape) marked with the letter **E**. Information systems used to support processes at runtime are depicted as squares with rounded corners and marked with the letter **S**. Diagnostic information is denoted by a star shape marked with the letter **D**. We distinguish between conformance-related diagnostics (star shape marked with **CD**) and performance-related diagnostics (star shape marked with **PD**). The twenty atomic use cases can be chained together in so-called composite use cases. These composite cases can be used to describe realistic BPM scenarios.

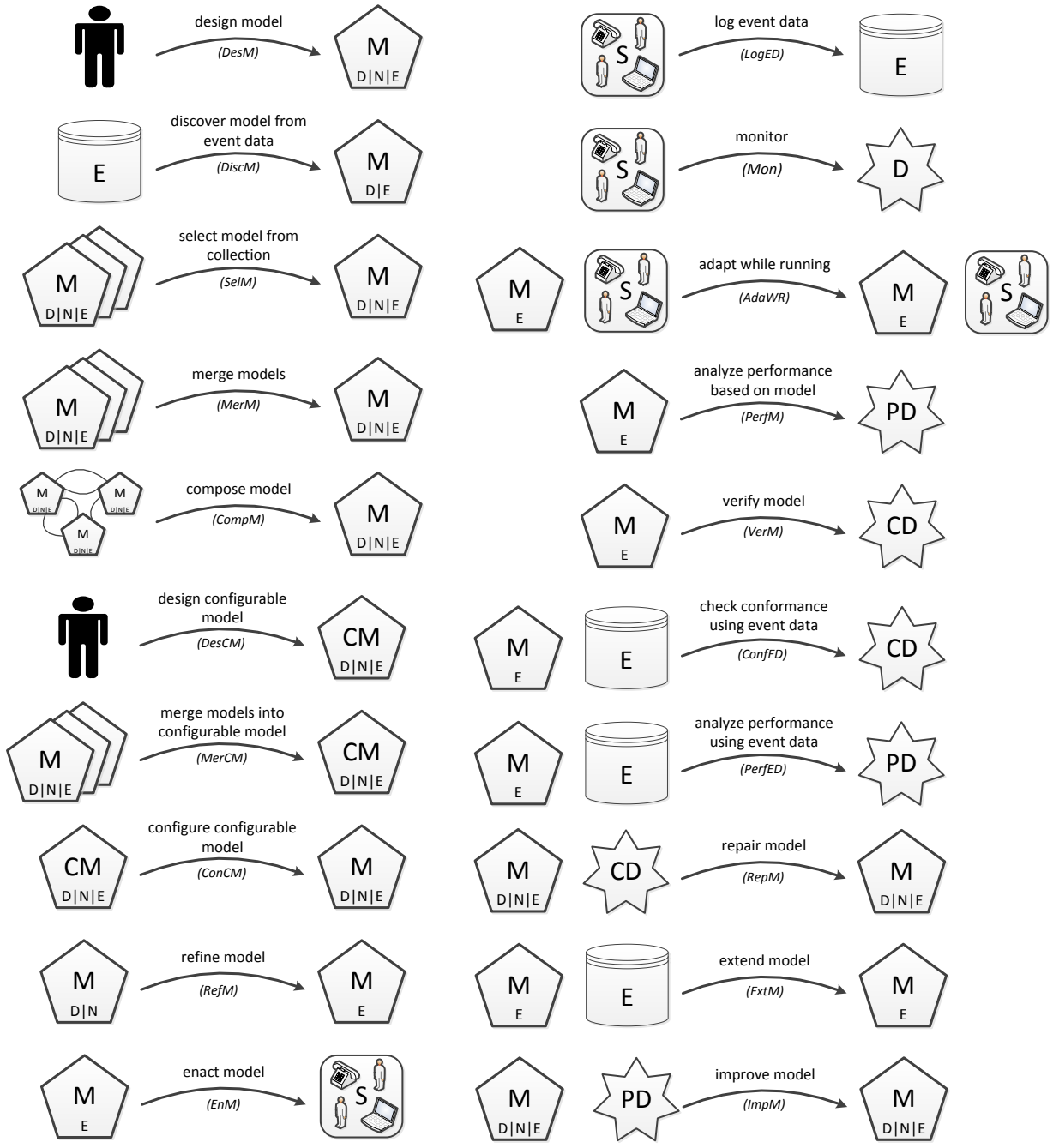


Figure 1: Twenty BPM use cases (Van der Aalst, 2013a). Use cases Log Event Data (LogED), Discover Model from Event Data (DiscM), Check Conformance Using Event Data (ConfED), Analyze Performance Using Event Data (PerfED), Repair Model (RepM), Extend Model (ExtM), Improve Model (ImpM) are most related to process mining. Use case Analyze Performance Based on Model (PerfM) includes traditional forms of simulation.

In (Van der Aalst, 2013a), the BPM literature is analyzed to see trends in terms of the twenty use cases, e.g., topics that are getting more and more attention. Here we only mention the use cases

most related to process mining.

- Use case *Log Event Data* (LogED) refers to the recording of event data, often referred to as event logs. Such event logs are used as input for various process mining techniques. XES (extensible event stream), the successor of MXML (mining XML format), is a standard format for storing event logs (www.xes-standard.org).
- Use case *Discover Model from Event Data* (DiscM) refers to the automated generation of a process model using process mining techniques. Examples of discovery techniques are the alpha algorithm, language-based regions, and state-based regions. Note that classical synthesis approaches need to be adapted since the event log only contains examples.
- Use case *Check Conformance Using Event Data* (ConfED) refers to all kinds of analysis aiming at uncovering discrepancies between modeled and observed behavior. Conformance checking may be done for auditing purposes, e.g., to uncover fraud or malpractices. Token-based (Rozinat and van der Aalst, 2008) and alignment-based (van der Aalst, Adriansyah, and van Dongen 2012) techniques replay the event log to identify non-conformance (Weerdt, De Backer, Vanthienen, and Baesens, 2011).
- Use case *Analyze Performance Using Event Data* (PerfED) refers to the combined use of models and timed event data. By replaying an event log with timestamps on a model, one can measure delays, e.g., the time in-between two subsequent activities. The results of timed replay can be used to highlight bottlenecks. Moreover, the gathered timing information can be used for simulation or prediction techniques (De Weerdt et al. 2012).
- Use case *Repair Model* (RepM) uses the diagnostics provided by use case ConfED to adapt the model such that it better matches reality. On the one hand, a process model should correspond to the observed behavior. On the other hand, there may be other forces

influencing the desired target model, e.g., a reference model, desired normative behavior, and domain knowledge.

- Event logs refer to activities being executed and events may be annotated with additional information such as the person/resource executing or initiating the activity, the timestamp of the event, or data elements recorded with the event. Use case *Extend Model* (ExtM) refers to the use of such additional information to enrich the process model. For example, timestamps of events may be used to add delay distributions to the model. Data elements may be used to infer decision rules that can be added to the model. Resource information can be used to attach roles to activities in the model (Rozinat, Wynn, van der Aalst, ter Hofstede, Fidge, 2009).
- Use case *Improve Model* (ImpM) uses the performance related diagnostics obtained through use case PerfED. ImpM is used to generate alternative process models aiming at process improvements, e.g., to reduce costs or response times. These models can be used to do ``what-if" analysis. Note that unlike RepM the focus ImpM is on improving the process itself.

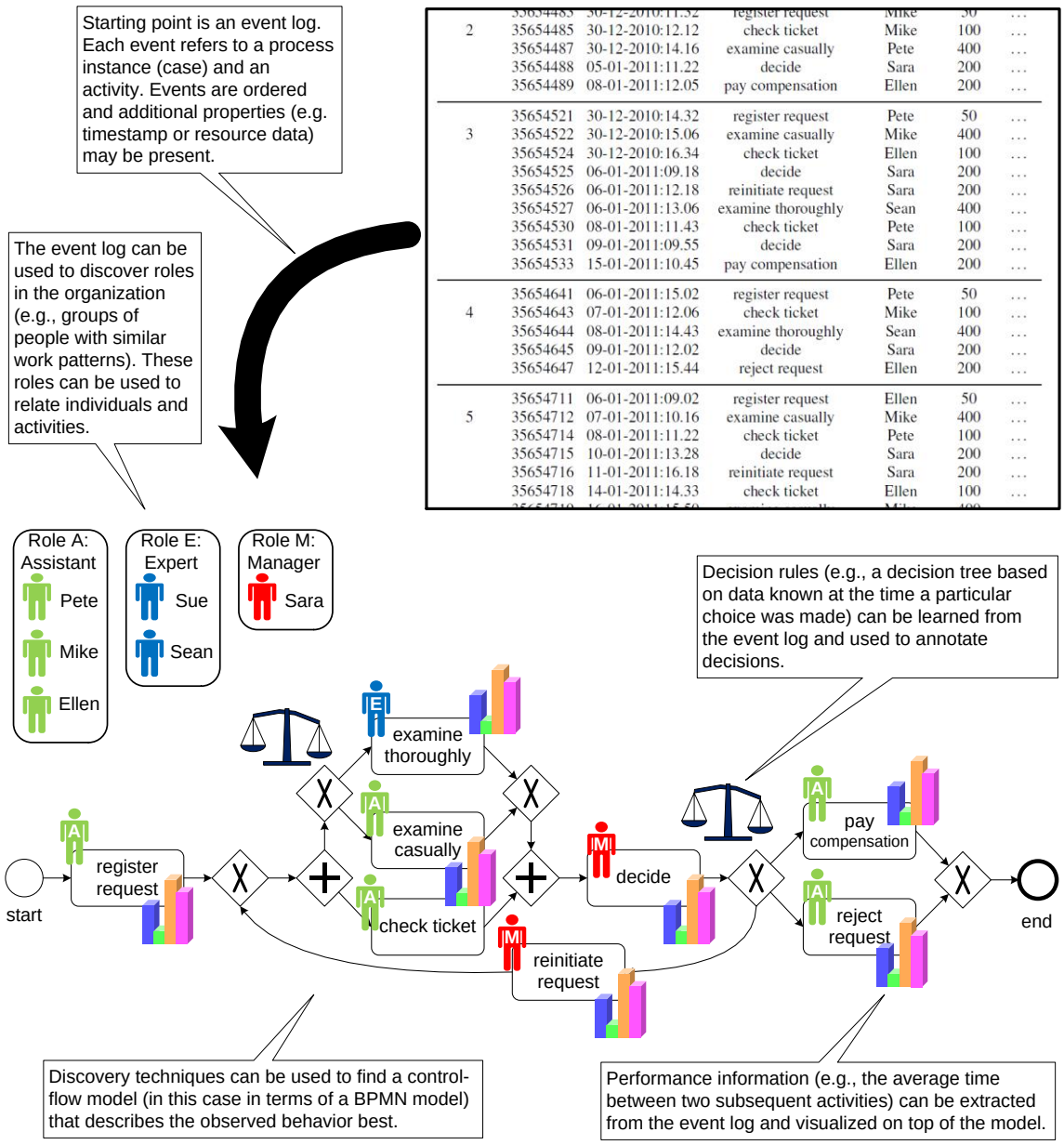
These use cases illustrate the different ways in which models and event data can be used. Use case Analyze Performance Based on Model (PerfM) includes traditional forms of simulation not directly driven by event data. See (van der Aalst 2010) for a discussion on the relation between simulation, BPM and process mining.

Process Mining

Over the last decade, process mining emerged as a new scientific discipline on the interface between process models and event data (van der Aalst, 2011). On the one hand, conventional

Business Process Management (BPM) and Workflow Management (WfM) approaches and tools are mostly model-driven with little consideration for event data. On the other hand, Data Mining (DM), Business Intelligence (BI), and Machine Learning (ML) focus on data without considering end-to-end process models. Process mining aims to bridge the gap between BPM and WfM on the one hand and DM, BI, and ML on the other hand.

The starting point for process mining is not just any data, but *event data* (IEEE Task Force on Process Mining 2012). Data should refer to discrete events that happened in reality. A collection of related events is referred to as 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. It is important to note that an event log contains only example behavior, i.e., we cannot assume that all possible runs have been observed. In fact, an event log often contains only a fraction of the possible behavior (van der Aalst, 2011). Frequently, event logs store additional information about events and these additional data attributes may be used during analysis. For example, many 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).



2	35654485	30-12-2010:12.12	check ticket	Mike	100	…
	35654487	30-12-2010:14.16	examine casually	Pete	400	…
	35654488	05-01-2011:11.22	decide	Sara	200	…
	35654489	08-01-2011:12.05	pay compensation	Ellen	200	…
3	35654521	30-12-2010:14.32	register request	Pete	50	…
	35654522	30-12-2010:15.06	examine casually	Mike	400	…
	35654524	30-12-2010:16.34	check ticket	Ellen	100	…
	35654525	06-01-2011:09.18	decide	Sara	200	…
	35654526	06-01-2011:12.18	reinitiate request	Sara	200	…
	35654527	06-01-2011:13.06	examine thoroughly	Sean	400	…
	35654530	08-01-2011:11.43	check ticket	Pete	100	…
	35654531	09-01-2011:09.55	decide	Sara	200	…
	35654533	15-01-2011:10.45	pay compensation	Ellen	200	…
4	35654641	06-01-2011:15.02	register request	Pete	50	…
	35654643	07-01-2011:12.06	check ticket	Mike	100	…
	35654644	08-01-2011:14.43	examine thoroughly	Sean	400	…
	35654645	09-01-2011:12.02	decide	Sara	200	…
	35654647	12-01-2011:15.44	reject request	Ellen	200	…
5	35654711	06-01-2011:09.02	register request	Ellen	50	…
	35654712	07-01-2011:10.16	examine casually	Mike	400	…
	35654714	08-01-2011:11.22	check ticket	Pete	100	…
	35654715	10-01-2011:13.28	decide	Sara	200	…
	35654716	11-01-2011:16.18	reinitiate request	Sara	200	…
	35654718	14-01-2011:14.33	check ticket	Ellen	100	…

Figure 2: Process mining techniques extract knowledge from event logs in order to discover, monitor and improve processes.

Event logs can be used to conduct various types of process mining, as is illustrated in Figure 2. Here we only mention the three main forms of process mining. 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 executions 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. Note that the three main forms of process mining correspond to some of the uses cases mentioned before.

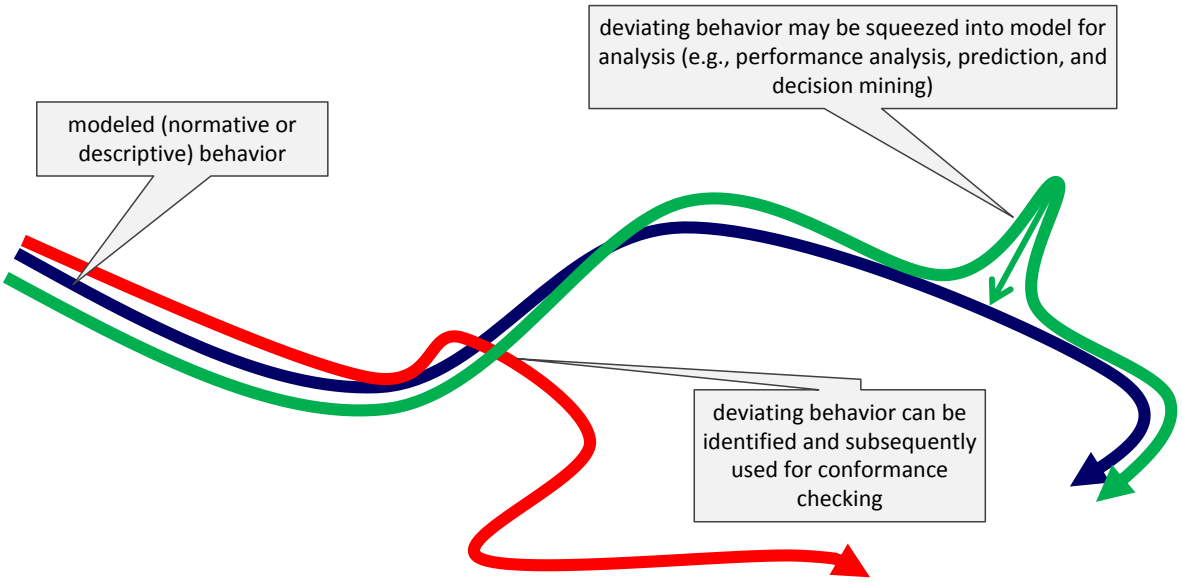


Figure 3: Process mining aligns observed and modeled behavior: "moves" seen in reality are related to "moves" in the model (if possible).

One of the key contributions of process mining is its ability to relate observed and modeled behavior at the event level, i.e., traces observed in reality (process instances in event log) are aligned with traces allowed by the model (complete runs of the model). As shown in Figure 3 it

is useful to align both even when model and reality disagree (van der Aalst, Adriansyah, and van Dongen 2012). First of all, it is useful to highlight where and why there are discrepancies between observed and modeled behavior. Second, deviating traces need to be "squeezed" into the model for subsequent analysis, e.g., performance analysis or predicting remaining flow times. The latter is essential in case of non-conformance (van der Aalst, Adriansyah, and van Dongen 2012). Without aligning model and event log, subsequent analysis is impossible or biased towards conforming cases.

Management of Large Process Model Collections

As organizations start to develop and maintain large collections of process models, there is an increasing need for continuous and efficient management of these process repositories. In order to reduce redundancy and improve maintainability of process model collections, efficient techniques to manage multiple process models in relation to each other as well as management of different versions of a single model are required. In (Dijkman, La Rosa, Reijers 2012), an overview of state-of-the-art management techniques for process model collections is provided. A pictorial representation of such techniques is provided in Figure 4.

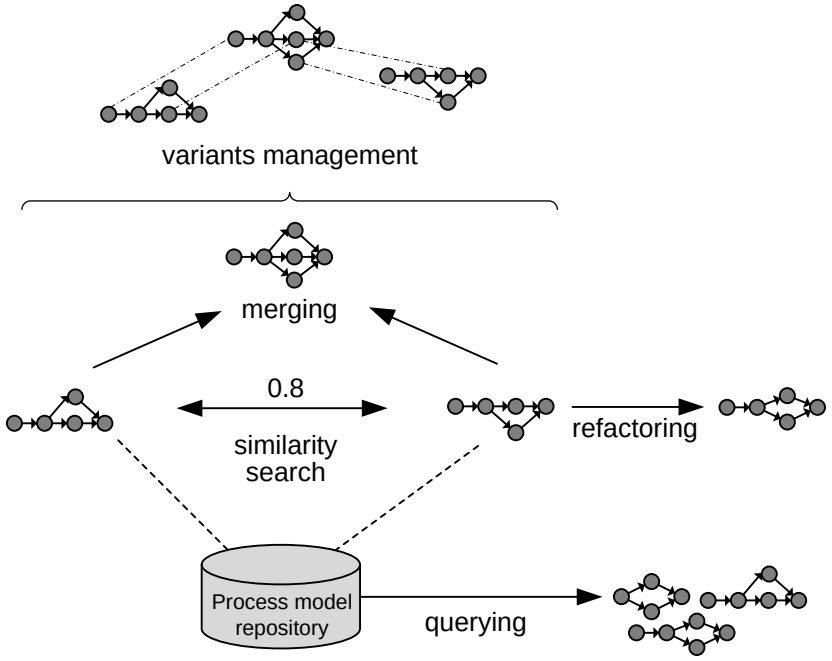


Figure 4: Overview of techniques for the management of process model collections (adapted from Dijkman, La Rosa, Reijers 2012)

As a collection of process models evolves over time, it may start to display unnecessary internal complexity. A common example is redundancy in the form of exact or approximate clones. Such clones are typically the result of copy/paste activities and they adversely affect the maintainability of process model collections, besides leading to unwanted inconsistencies in the repository, if they are modified independently of each other. Clones can manifest themselves both at the level of entire process models as well as fragments thereof. Researchers have proposed various techniques for detecting such clones within process model repositories (see e.g. Guo, Zou 2008; Dumas, García-Bañuelos, La Rosa, Uba 2013, Ekanayake, Dumas, García-Bañuelos, La Rosa, ter Hofstede 2012). *Refactoring* techniques, inspired from software engineering, have been explored to improve the maintainability and readability of process model collections. Examples of such refactoring techniques are extracting the identified clones and storing them as reusable sub-processes (Dumas, García-Bañuelos, La Rosa, Uba 2013),

standardizing approximate clones (Ekanayake, Dumas, García-BañuelosLa Rosa, ter Hofstede 2012), and modularizing process models into different levels of abstraction (Weber, Reichert, Mendling, Reijers 2011, Dijkman, Gfeller, Kuster, Volzer 2011).

Another set of management techniques relate to the notion of *similarity search* (Becker and Laue 2012). Given a collection of process models and a search process model, similarity search techniques identify and return those models from the collection that are deemed similar (e.g., potentially inexact matches) to the search model. A potential use case of similarity search is for an organization to identify which of its own processes are similar to a standardized (reference) process model. Research in this area approaches this challenge from two angles: i) the definition and implementation of similarity measures that return a similarity rating (e.g., between 0 and 1) for two process models (Dijkman, Dumas, van Dongen, Kaarik, Mendling 2011); and ii) the implementation of indexing techniques for improving the retrieval of similar process models (Yan, Dijkman, Grefen 2010, Kunze, Weske, 2011) or of models containing the query model as a sub-graph (e.g. Jin, Wang, La Rosa, ter Hofstede, Wen, 2012). More in general, such indexing techniques can be used as the backbone for efficient *querying* of process model repositories. The query could be expressed as a model (e.g. Jin, Wang, La Rosa, ter Hofstede, Wen, 2012, Awad, Sakr 2012) or in textual form (e.g. Jin, Wang, Wen 2011).

Process model merging is concerned with merging a collection of process variants into one consolidated process model which can be very useful in the context of organizational mergers, restructurings and rationalizations. This can lead to a collection of reduced size that has been standardized and optimized for the current business context, which in turn can significantly improve the maintainability of the collection as a whole. Some merging techniques enforce behavior-preservation such that the merged model maintains the behavior of all individual

models allowing one to replay the behavior of each input variant on the merged model (La Rosa et al., 2013). Some of the merging techniques take into account notions of label similarities when merging so that it is possible to merge activities from different models that have similar but not identical labels (Gottschalk, van der Aalst, Jansen-Vullers, 2008; La Rosa et al., 2010) whereas others only merge activities with identical labels (Reijers, Mans, van der Toorn, 2009; Sun, Kumar, Yen, 2006; Mendling, Simon, 2006).

Given a collection of process models, *variants management* mechanisms (e.g., Pascalau, Awad, Sakr, Weske, 2010, Ekanayake, La Rosa, ter Hofstede, Fauvet, 2011, Weidlich, Mendling, Weske, 2011) are required to keep track of the organization of the collection such that users can browse it and view its evolution as changes are made. These mechanisms are based on various types of relations between the process models of the repository. For example, Ekanayake, La Rosa, ter Hofstede and Fauvet (2011) exploit information on shared clones across process models and versions thereof, in order to provide change propagation and access control features. Other common relations that can be used to manage variants are aggregation and generalization relations (Kurniawan, Ghose, Le, Dam 2012). An aggregation relation exists between a business process model and it parts (usually sub-processes) while a generalization relation exists between a more general process model and a more specific one. Aggregation and generalization are typically used to develop a hierarchical classification of processes models, enabling users to navigate a collection of process models by traversing the hierarchy.

While there is a plethora of techniques for managing large process model collections, a collection remains “static” in most cases until a process improvement initiative is carried out. This is an area where considerable progress can still be made towards “self-healing” or “self-adapting” process model collections.

Research Innovations

We propose to integrate process mining and the management of process model collections, leading to a solution for the management of liquid process model collections. This results in the following innovations:

- **Innovation 1: *Confidence → Evidence***. To address Issue 1 mentioned in the introduction (process model collections based on subjective perceptions), we propose to create a new entry point to the process lifecycle. Traditionally, business processes are first designed and then executed on the basis of these designs, which are informed by domain expertise (*confidence-based process design*). This has proven to be time-consuming and error-prone since organizational behavior is inherently difficult to capture formally. This problem is exacerbated in the context of large firms executing many complex and disparate business processes. We propose to leverage off the organizational knowledge stored in event logs to provide an alternative starting point to process model design (*evidence-based process design*). By developing semi-automated techniques, this design activity will be less *expensive* and result in *more current* process models.
- **Innovation 2: *Static → Dynamic***. In response to Issue 2 (process model collections out of-synch with organizational behavior), we challenge the *static* nature of process model collections, manually built once, and updated from time to time. Rather, we propose *liquid* process model collections as a *dynamic* artifact that can self-adapt to evolving business operations, as recorded in logs, thereby consistently remaining current and relevant. This *continuous realignment* of process model collections with organizational behavior can virtually eliminate the risk of obsolete process models and concomitantly increase the value of BPM initiatives. As opposed to traditional process monitoring and

controlling, it will be possible to adapt process model collections based on event logs, even if their constituent process models are not automatically enacted by a Business Process Management System.

- **Innovation 3: *Generic → Demand-driven***. Today's process model collections are "common ground": they are inspired by *generic* business needs and designed before specific stakeholder demands are articulated. As such, they are problem-independent and user-independent. With the availability of liquid process model collections, which *incorporate execution data* inferred from event logs, such as actual process performance metrics or resource allocations, it will be possible to generate insightful reports, the result of which will be new, *demand-driven* process models tailored to the needs of specific stakeholders. This will address Issue 3 (process model collections designed for the most common purposes).

These innovations result in the so-called "liquid" business process model/log collection mentioned in Section 1.

Realization

In order to realize the above innovations, we propose a research agenda consisting of five interrelated research streams (RS1-RS5), which are illustrated in Figure 5. Each stream aims to realize one or more of the innovations identified in Section 5. Details for each stream are provided below.

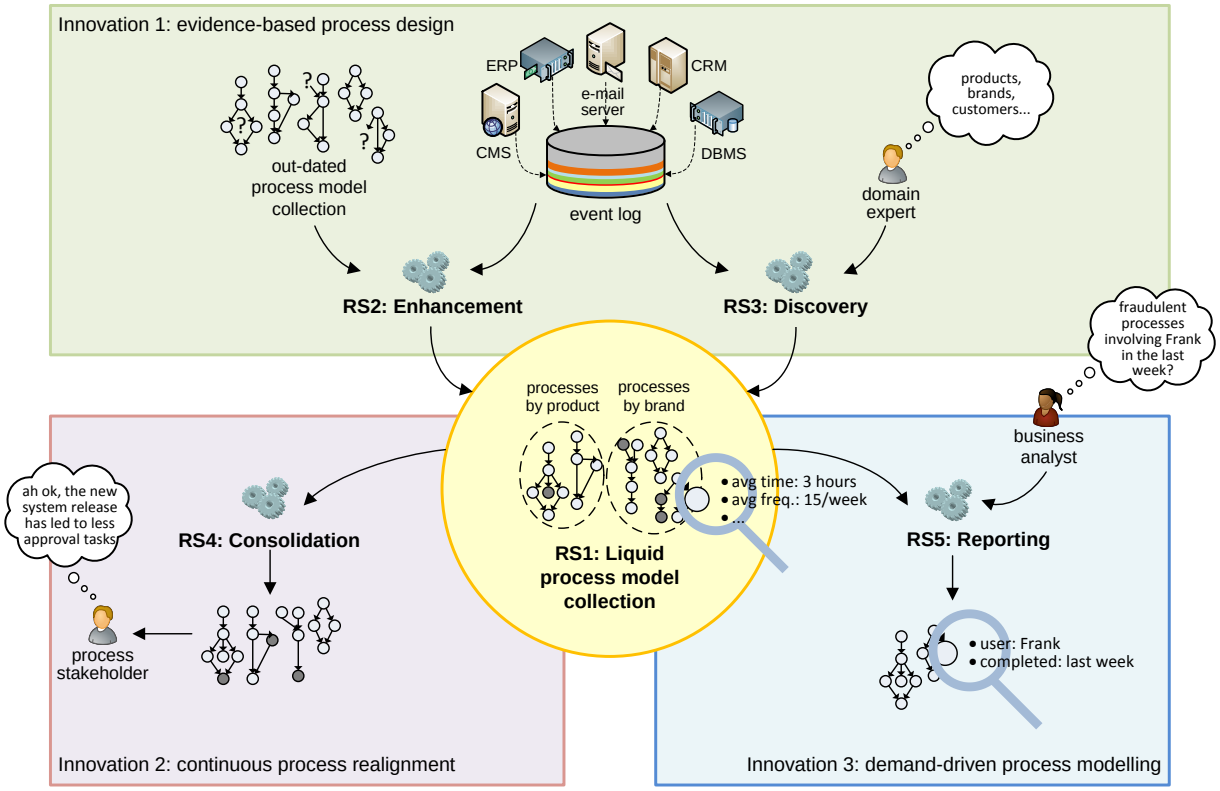


Figure 5: **The proposed research agenda consists of five research streams (RS1-RS5), each mapped to one or more innovations.**

Research Stream 1 (RS1): Fundamentals of liquid process model collections. This research stream focusses on the foundational notions behind *liquid process model collections* and techniques for operationalizing them. As such, RS1 is the enabler for the realization of Innovations 1-3 through the other research streams. The first priority in this stream should be the extension of the concept of *alignment* between a log and a *single* process model (van der Aalst, Adriansyah, and van Dongen 2012, Adriansyah et al. 2011) to the realm of process model collections, in order to determine an *overall alignment score* between logs and process model collections. This notion, illustrated in Figure 6, will need to consider different ways of partitioning the log, e.g. along entire traces or portions thereof (i.e. sub-traces), in order to

identify the sequence of events that best matches an individual process model. Further, for each model in the collection its execution occurrence frequency (i.e. number of cases) needs to be considered, as inferred from the logs, in order to weigh individual alignments.

Existing process discovery (van der Aalst 2011) and graph matching (Bunke 1997) algorithms may serve as the basis for a novel algorithm to compute the overall alignment score. The process mining environment ProM can be extended to incorporate these algorithms.

RS1 should also identify suitable *execution properties* inferred from the logs (e.g. performance indicators, bottlenecks and resource utilization) that can be linked to different elements of the process model collection to enrich it (e.g. the average duration of single tasks or the frequency of a whole process). The types of relations should be captured in the form of a *conceptual model*. We need to formalize a *data structure* to persist both the alignment and these properties for subsequent business analysis (proposed to be realized as part of RS5). Finally, a *customizable dashboard* needs to be designed and implemented, which uses this data structure to visualize relevant execution properties at various levels of abstraction and allows one to navigate across these levels. Such a dashboard can be implemented on top of the Apromore repository.

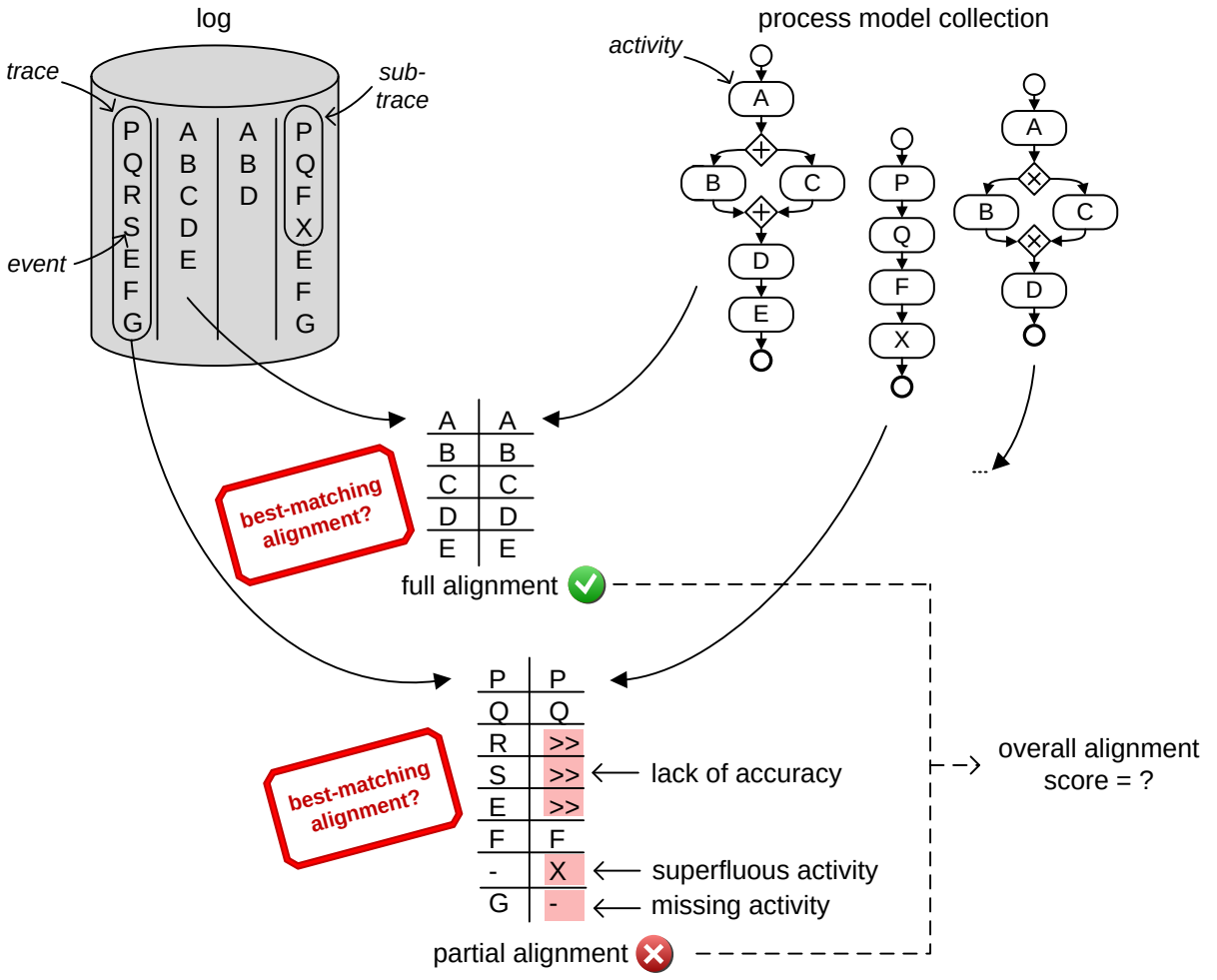


Figure 6: The backbone of liquid process model collections is a notion of overall alignment score between logs and process model collections.

Research Stream 2 (RS2): Enhancement of static process model collections. This stream aims to make existing, static process models liquid, thus contributing to the realization of Innovation 1 (evidence-based process design). This involves devising and implementing techniques for applying appropriate changes to an existing process model collection in order to improve its alignment (as defined in RS1) with the organizational behavior recorded in the log. Genetic algorithms such as simulated annealing (Suman 2004) can be leveraged for this purpose,

to apply perturbations to the models in the collection that keep change to a minimum.

In this stream we will also develop a method for enriching an aligned process model collection with log-inferred execution properties, as identified in RS1. Once again, the techniques envisaged in this research stream can be realized on top of ProM.

Research Stream 3 (RS3): Domain-driven discovery of liquid process model collections. This stream, complementary to RS2, aims to develop a set of parametric algorithms for the semi-automatic discovery of liquid process model collections from scratch. Therefore, it contributes to the realization of Innovation 1 (evidence-based process design).

The novelty of these algorithms is that they will operate over process model collections, rather than single models, and be *driven by domain knowledge*, which can be provided by subject-matter experts via parameters. In fact, while a plethora of process discovery algorithms is available (van der Aalst 2011), none of them allows domain knowledge to influence the discovery. Different algorithms should be developed based on the possible types of input. Examples of different types of input are company-specific dimensions (e.g. products, brands, customer types) as well as quality metrics mandated by the company (e.g. a threshold for process model size). Findings from multi-database mining (Wu et al. 2005) can potentially inform the envisaged algorithms in order to discover process model collections from heterogeneous logs, i.e. logs generated from various systems. Finally, mechanisms should be developed so that the discovered collections can be enriched with execution properties relevant to the domain knowledge used as input. These mechanisms should be based on the techniques defined in RS2.

This stream also aims to develop a technique for inferring relationships between the discovered

process models (e.g. abstraction and order relationships) and use these to compose a process architecture (Eid-Sabbagh et al. 2012) to accompany the collection. Similar to RS2, the discovery algorithms and the technique for inferring process architectures can be implemented in ProM.

Research Stream 4 (RS4): Consolidation of liquid process model collections. This stream aims to design and implement an approach for retaining model alignment with organizational behavior, as the latter evolves over time. As such, this stream contributes to the realization of Innovation 2 (continuous process realignment).

A challenge to achieve this continuous realignment will be how to distinguish transient changes in organizational behavior from those of a more permanent nature, aka *concept drifts* (Bose et al. 2011), that need to be reflected onto the process model collection. Techniques for process model change required to realign a collection should be informed by RS2, and extended to deal with updates in the associated execution properties. Conversely, a mechanism should be developed to recognize changes intentionally made to the process model collection that may have not yet been observed in the log. This situation is illustrated in Figure 7.

Another component of this research stream involves the design of a console that notifies process stakeholders about potential changes to the process models of the collection that concern them, and which can visualize these changes as delta-differences on top of the existing models. Such a console will assist the stakeholders with change assessment, i.e. deciding whether and if so, to what degree to commit changes. For example, one may decide to only incorporate those changes that have been observed relatively frequently as part of new organizational behavior. This console should also allow stakeholders to provide feedback on those changes that are considered

undesirable. This may provide input to targeted analyses, which is the scope of RS5.

Further, in this stream process model version control mechanisms (Ekanayake et al. 2011) should be extended so that one can keep track of all versions of process model collections and their execution properties by storing delta-differences. This may provide input to targeted analyses in RS5, e.g. by comparing changes that have occurred over given timeframes.

The consolidation approaches and console can be implemented in Apromore.

Research Stream 5 (RS5): Reporting for liquid process model collections. This stream aims to develop techniques to generate *sophisticated reports* on a liquid process model collection. Such reports should cover historical organizational behavior (descriptive nature) or forecasts of future organizational behavior (predictive nature).

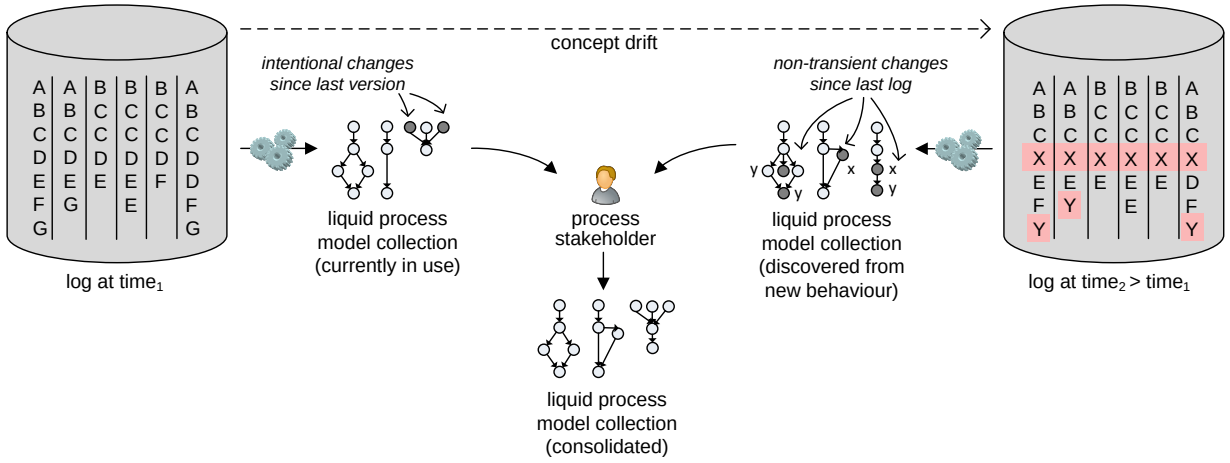


Figure 7: Consolidating a process model collection in order to cope with evolutions of organizational behavior (e.g. due to new laws).

These reports are likely to provide deep insights into different organizational aspects, ranging from performance issues to compliance violations and fraudulent activities. For example, it will be possible to generate all process models in which the employee Frank was involved, in the last month, that lasted 10 days; or all process models affected by a fraud through a change of account

in the last year; or all process models that are likely to be impacted by the bankruptcy of an important client in the next month.

The results of these reports are presented in the form of new process models, driven by specific stakeholders' demands, and serve as the basis for more informed organizational decisions. As such, this stream aims to realize Innovation 3 (demand-driven process modeling).

Process mining techniques are able to reveal the reasons for a good or bad performance and may even provide predictions and recommendations. However, for "what-if" analysis we will often need to resort to simulation. See Van der Aalst (2010) for a discussion on the interplay between simulation and process mining.

This stream also involves the design and development of a query language exploiting the data structure defined in RS1. This language is to be supported by a dashboard that business analysts can use to submit their queries.

Indexing techniques from graph databases (Jin et al. 2013) will need to be extended to index specific elements of a liquid process model collection (e.g. execution properties) in order to efficiently execute queries over large, property-rich collections, while statistical techniques (e.g. regression analysis – Freedman 2005) and data mining algorithms (e.g. decision tree building algorithms – Quinlan 1993) can be employed as a basis for generating process model forecasts. The liquidity property of the collection will guarantee that the results of the queries will always be relevant.

The envisioned dashboard and the underlying reporting techniques can be implemented in Apromore.

Supporting Software

To realize the intended innovations we will build on ProM and Apromore and tightly integrate both. ProM is a generic open-source framework for implementing process mining tools in a standard environment. It can be downloaded from www.processmining.org. The ProM framework can load event logs in standard formats such as XES and MXML. The ProM toolkit has been around for about a decade. During this period, the ProM framework has matured to a professional level. Dozens of developers in different countries contributed to ProM in the form of plug-ins. In the current version more than 600 plug-ins are available distributed over 100 packages that can be loaded separately. Through these plug-ins ProM supports the entire process mining spectrum:

- **Online and off-line process mining**. Event data can be partitioned into "pre mortem" and "post mortem" event logs. The term "post mortem" event data refers to information about cases that have completed, i.e., these data can be used for process improvement and auditing, but not for influencing the cases they refer to. The term "pre mortem" event data refers to cases that have not yet completed. If a case is still running, i.e., the case is still "alive" (pre mortem), then it may be possible that information in the event log about this case (i.e., current data) can be exploited to ensure the correct or efficient handling of this case. "Post mortem" event data is most relevant for off-line process mining, e.g., discovering the control-flow of a process based on one year of event data. For online process mining, mixtures of "pre mortem" (current) and "post mortem" (historic) data are needed. For example, historic information can be used to learn a predictive model. Subsequently, information about a running case is combined with the predictive model to provide an estimate for the remaining flow time of the case.

- **Different model types.** Two types of models can be identified: "de jure models" and "de facto models". A de jure model is normative, i.e., it specifies how things should be done or handled. For example, a process model used to configure a BPM system is normative and forces people to work in a particular way. A de facto model is descriptive and its goal is not to steer or control reality. Instead, de facto models aim to capture reality. Both de jure and de facto models may cover multiple perspectives including the control-flow perspective ("How?"), the organizational perspective ("Who?"), and the case perspective ("What?"). These are supported by ProM. The control-flow perspective describes the ordering of activities. The organizational perspective describes resources (worker, machines, customers, services, etc.) and organizational entities (roles, departments, positions, etc.). The case perspective describes data and rules. Process mining can be used to determine the degree to which organizational aspects, as modeled in the above process perspectives, conform to observed behavior.
- **Different types of process mining**. As discussed before, process mining includes discovery, conformance checking, and enhancement. However, also more advanced forms of process mining like prediction, recommendation, and concept-drift analysis are supported by ProM.

Currently, ProM does not support the management of collections of models and logs.

Apromore is an open and extensible repository to store and disclose business process models of a variety of languages, such as BPMN, eEPCs, YAWL and Workflow nets. Apromore provides state-of-the-art features to facilitate the management of large process model collections.

The backbone of Apromore is a fragment-based version control mechanism (Ekanayake et al. 2011). Accordingly, each single-entry single-exit (SESE) fragment of each process model is

independently versioned, and can be associated with an owner. This mechanism allows automatic detection of cloned fragments both between different versions of the same process model as well as between different process models. Cloned fragments are only stored once (thus allowing "vertical sharing" between different versions of the same process model, and "horizontal sharing" between different process models).

Sharing fragments vertically allows Apromore to easily track differences between versions of the same model. Sharing fragments horizontally enables change propagation features. When one makes a change to a process model fragment, the owners of all process models in the repository that will be impacted by this change (because these models contain clones to the fragment being changed) will be notified. Depending on the change propagation policy, one can decide to commit the change, or not (in the latter case, a separate version of that fragment will be created).

This fragment-based version control mechanism also allows access control at the level of single fragments, and concurrency control: multiple users can simultaneously work even on the same process model, provided they edit unrelated fragments.

Moreover, like in software code, process model versions are organized in branches, where one branch may be created by "branching out" from a version in an existing branch.

Besides this fragment-based version control mechanism, Apromore provides a wealth of features to manage large process model models, such as, for example, searching for similar models, merging similar models into a consolidated process model, identifying clones within and across process models, and cluster them.

Apromore relies on an internal format called *canonical process format* to support the above features. Whether one is after finding the similarity between your process models, or detecting

clones, Apromore performs all these operations on the canonical format of the process models stored in its repository. This way one can, for example, compare process models defined in different languages such as EPCs and BPMN, or merge them into a process model and then decide the target language for this new model.

The canonical process format provides a common, unambiguous representation of business processes captured in different languages and/or at different abstraction levels, such that all process models can be treated alike.

Apromore is available as a Sofware as a Service (SaaS) at www.apromore.org. It relies on a plugin framework based on OSGi. This way, new features can be added in the form of OSGi plugins on-the-fly, and similarly, existing features can be uninstalled without the need to restart Apromore.

Presently, Apromore is tailored towards the management of process models rather than event logs. Thus, the tool does not provide any feature to align modeled and observed behavior. In order to address the challenges mentioned above, a tight integration between Apromore and ProM is needed. Moreover, we also envision the integration of other tools. For example, from ProM we can call the simulation CPN Tools for large-scale simulation experiments.

Conclusion

This chapter discussed the need to *relate modeled and observed behavior for large collections of processes*. After introducing the main BPM use cases, we discussed the state-of-the-art in process mining and managing large process model collections. The “disconnect” between process mining research and the management of large model collections is severely limiting the application of BPM technologies. Therefore, we suggest three major research innovations and

five different research streams to realize these innovations. The aim is to create “liquid” business process model collections, i.e., collections of process models that are synchronized with the organizational reality and continuously adapt to evolving circumstances. This is the only way to breathe life into business process model collections. Without it, model collections will be static, and thus of limited value; they will soon become outdated unless they are manually updated, which is often an expensive operation.

In this chapter we invite the research community to contribute to the research streams identified in this chapter, and hope that we made a good case that Apromore and ProM provide a good starting point for realizing the ideas presented. As described in the five research streams, the challenges that need to be dealt with are manifold, ranging from the continuous alignment of models and event logs to refactoring of process model collections with event data and advanced reporting on such collections.

In Ekanayake et al. (2013) we did some initial work in the direction of bridging this gap between process mining and management of large process model collections. Specifically, we developed a technique on top of Apromore and ProM, called *Slice, Mine & Dice (SMD),* that can mine a hierarchical collection of process models from an event log, where each model in the collection has a bounded complexity (e.g. on the model size) that can be set by the user. Further, in Van der Aalst (2013c), we introduced the *process cube* notion. The process cube structures event data using different dimensions (type of process instance, type of event and time window) in order to discover multiple inter-related processes or check the conformance thereof. Moreover, by precisely aligning event data and process models we enable new types of simulation (Van der Aalst 2010, 2013b) where real and simulated behaviors are combined. These ideas illustrate that “liquid” business process model collections are likely to trigger new forms of process

management.

References

–, CXO "The importance of BPM in a fast changing business environment". Issue 9, April 2008. www.cxo.eu.com (accessed: Feb 2013).

–, eBizQ, "For many businesses, BPM ranks as a top priority for 2012". 02/01/2012. www.ebizq.net (accessed: Feb 2013).

–, WinterGreen Research, "Business Process Management (BPM) Market Shares, Strategies, and Forecasts Worldwide 2012 to 2018", 2012.

Adriansyah, A., van Dongen, B., and van der Aalst, W.M.P. "Conformance Checking using Cost-Based Fitness Analysis", *Proceedings of EDOC*, IEEE Computer Society, 2011

Awad, A., Sakr, S: On efficient processing of BPMN-Q queries. *Computers in Industry* 63(9): 867-88, 2012.

Becker, M., Laue, R.: A comparative survey of business process similarity measures. *Computers in Industry* 63(2): 148-167, 2012.

Bose, R.P.J.C., van der Aalst, W.M.P., Zliobaite, I., and Pechenizkiy, M. "Handling Concept Drift in Process Mining", *Proceedings of CAiSE*, LNCS 6741, Springer, 2011

Bunke, H. "On a relation between graph edit distance and maximum common subgraph", *Pattern Recognition Letters* (18:8), 1997.

Davies, I., Green, P., Rosemann, M., Indulska, M., and Gallo, S. "How do Practitioners Use Conceptual Modeling in Practice?", *Data & Knowledge Engineering* (58:3), 2006.

Dijkman, R.M., Dumas, M., van Dongen, B.F., Käärik, R., and Mendling, J., "Similarity of business process models: Metrics and evaluation", *Information Systems* (36:2), 2011

Dijkman, R.M., La Rosa, M., and Reijers, H.A. "Managing Large Collections of Business Process Models – Current Techniques and Challenges", *Computers in Industry*, (63:2), 2012.

Dumas, M., García-Bañuelos, L., La Rosa, M., Uba, R.: Fast detection of exact clones in business process model repositories. *Inf. Syst.* 38(4): 619-633, 2013.

Dumas, M., La Rosa, M., Mendling, J. and Reijers, H.A. *Fundamentals of Business Process Management*, Springer, 2013.

Eid-Sabbagh, R.-H., Dijkman, R.M., and Weske, M. "Business Process Architecture: Use and Correctness", *Proceedings of BPM*, LNCS 7481, Springer, 2012.

Ekanayake, C.C., Dumas, D., García-Bañuelos, L., La Rosa, M: Slice, Mine and Dice: Complexity-Aware Automated Discovery of Business Process Models. *Proceedings of BPM*, LNCS 8094: 49-64, 2013.

Ekanayake, C.C., Dumas, D., García-Bañuelos, L., La Rosa, M., ter Hofstede, A.H.M.: Approximate Clone Detection in Repositories of Business Process Models. *Proceedings of BPM*: 302-318, 2012.

Ekanayake, C.C., La Rosa, M., ter Hofstede, A.H.M., and Fauvet, M.C., "Fragment-Based Version Management for Repositories of Business Process Models", *Proceedings of CoopIS*, LNCS 7044, Springer, 2011.

Fahland, D., and van der Aalst, W.M.P, "Repairing Process Models to Reflect Reality", *Proceedings of BPM*, LNCS 7481, Springer, 2012.

Freedman, D.A., *Statistical Models: Theory and Practice*, Cambridge University Press, 2005.

Gotts, I., "Putting the M back in BPM", *BPTrends*, 2010. www.bptrends.com (accessed: Feb 2013).

Gottschalk, F. van der Aalst, W.M.P. Jansen-Vullers,M.H. Merging event-driven process chains, in: Proc. of CoopIS, vol. 5331 of LNCS, Springer, 2008, pp. 418–426.

Hilbert, M. and Lopez, P., The World's Technological Capacity to Store, Communicate, and Compute Information. Science, 332(6025), 60-65, 2011.

IEEE Task Force on Process Mining. Process Mining Manifesto. In F. Daniel, K. Barkaoui, and S. Dustdar, editors, Business Process Management Workshops, volume 99 of Lecture Notes in Business Information Processing, pages 169-194. Springer-Verlag, Berlin, 2012.

Jin, T., Wang, J., Wu, N., La Rosa, M., and ter Hofstede, A.H.M. "Efficient Querying of Large Process Model Repositories", *Computers in Industry*, (64:1), 2013.

Kunze, M., Weske, M.: Metric Trees for Efficient Similarity Search in Large Process Model Repositories. Business Process Management Workshops 2010: 535-546, 2011.

La Rosa, M., Dumas, M., Uba, R. and Dijkman, R.M., Business Process Model Merging: An Approach to Business Process Consolidation, *ACM Transactions on Software Engineering and Methodology* (22:2), 2013.

Mendling, J. Simon, C. Business process design by view integration, in: Proc. Of BPM Workshops, vol. 4103 of LNCS, Springer, 2006: 55–64.

Pascalau, E., Awad, A., Sakr, S., Weske, M.: On Maintaining Consistency of Process Model Variants. Business

Process Management Workshops 2010: 289-300, 2011.

Quinlan, J.R., *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., 1993.

Raduescu, C., Tan, H. M., Jayaganesh, M., Bandara, W., zur Muehlen, M., and Lippe, S. "A Framework of Issues in Large Process Modeling Projects," *Proceedings of ECIS*, Association for Information Systems, 2006.

Reijers, H.A. Mans, R.S. van der Toorn, R.A. Improved model management with aggregated business process models, *Data and Knowledge Engineering* 68 (2), 2009: 221–243.

Reijers, H.A., Mendling, J., and Dijkman, R.M., "Human and automatic modularizations of process models to enhance their comprehension", *Information Systems* (36:5): 2011.

Rozinat, A. and van der Aalst, W.M.P. Conformance Checking of Processes Based on Monitoring Real Behavior. Information Systems, 33(1), 64-95, 2008.

Rozinat, A., Wynn, M., van der Aalst, W.M.P., ter Hofstede, A.H.M., Fidge, C., Workflow simulation for operational decision support. Data and Knowledge Engineering. 68(9): 834-850, 2009.

Rosemann, M. "Potential Pitfalls of Process Modeling: Part B," *Business Process Management Journal* (12:3), 2006.

Suman, B. "Study of simulated annealing based algorithms for multi-objective optimization of a constrained problem", *Computers & Chemical Engineering*, (28:9), 2004.

Sun, S. Kumar, A. Yen,J. Merging workflows: a new perspective on connecting business processes, Decision Support Systems 42 (2) (2006) 844–85

Jin, T., Wang, J., La Rosa, M., ter Hofstede, A.H.M., Wen, L.: Efficient querying of large process model repositories. *Computers in Industry* 64(1): 41-49, 2013.

Jin, T., Wang, J., Wen, L: Querying Business Process Models Based on Semantics. DASFAA (2) 2011: 164-178.

van der Aalst, W.M.P., Business Process Management: A Comprehensive Survey. ISRN Software Engineering, pages 1-37, 2013a. doi:10.1155/2013/507984.

van der Aalst, W.M.P. Business Process Simulation Survival Guide. BPM Center Report BPM-13-11, BPMcenter.org, 2013b.

van der Aalst, W.M.P.: Process Cubes: Slicing, Dicing, Rolling Up and Drilling Down Event Data for Process Mining, *Proceedings of AP-BPM*, LNBIP 159, 1-22, Springer, 2013c.

van der Aalst, W.M.P., *Process Mining: Discovery, Conformance and Enhancement of Business Processes*,

Springer, 2011.

van der Aalst, W.M.P., Business Process Simulation Revisited. In J. Barjis, editor, Enterprise and Organizational Modeling and Simulation, volume 63 of Lecture Notes in Business Information Processing, pages 1-14. Springer-Verlag, Berlin, 2010.

van der Aalst W.M.P., Adriansyah, A., and van Dongen B., Replaying History on Process Models for Conformance Checking and Performance Analysis. WIREs Data Mining and Knowledge Discovery, 2(2):182-192, 2012.

van Dongen, B.F., Alves de Medeiros, A.K., and Wen, L., "Process Mining: Overview and Outlook of Petri Net Discovery Algorithms", *Transactions on Petri Nets and Other Models of Concurrency 2*, Springer, 2009.

De Weerdt, J., De Backer, M., Vanthienen, J., Baesens, B., A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. Information Systems 37(7):654-676, 2012.

Weber, B. Reichert, M. Mendling, J. Reijers, H.A. Refactoring large process model repositories, *Computers in Industry* 62 (5), 2011. 467–486.

Weidlich, W., Mendling, J., Weske, M.: A Foundational Approach for Managing Process Variability. CAiSE 2011: 267-282, 2011.

Wolf, C. and Harmon, P., "The State of Business Process Management 2012", *BPTrends*, 2012. www.bptrends.com (accessed: Feb 2013).

Wu, X., Zhang, C., Zhang, S., "Database classification for multi-database mining", *Information Systems* (30:1), 2005.

Yan, Z. Dijkman, R. Grefen, P. Fast business process similarity search with feature based similarity estimation, in: On the Move to Meaningful Internet Systems: OTM 2010, vol. 6426 of LNCS, Springer, 2010, pp. 60–77.

Prof.dr.ir. Wil van der Aalst is a full professor of Information Systems at the Technische Universiteit Eindhoven (TU/e). He is also the Academic Supervisor of the International Laboratory of Process-Aware Information Systems of the National Research University, Higher School of Economics in Moscow. Moreover, since 2003 he has a part-time appointment at Queensland University of Technology (QUT). At TU/e he is the scientific director of the Data

Science Center Eindhoven (DSC/e). His personal research interests include process mining, Petri nets, and business process management. Many of his papers are highly cited (he has an H-index of more than 107 according to Google Scholar). In 2012, he received the degree of doctor honoris causa from Hasselt University. In 2013, he was appointed as Distinguished University Professor of TU/e and was awarded an honorary guest professorship at Tsinghua University. He is also a member of the Royal Holland Society of Sciences and Humanities and the Academy of Europe.

Marcello La Rosa is associate professor and the Information Systems School Academic Director for corporate engagements at Queensland University of Technology in Brisbane, Australia. He is also a Researcher at the National ICT Australia. His research interests focus on process consolidation, configuration, mining and automation. Marcello has published over 60 refereed papers on these topics including papers in top journals like ACM TOSEM, Formal Aspects of Computing and Information Systems. He leads the Apromore initiative (www.apromore.org) – a strategic collaboration between various universities for the development of an advanced process model repository. Marcello has taught Business Process Management (BPM) to students and practitioners in Australia for over eight years. He is co-author of "Fundamentals of Business Process Management" (Springer, 2013), the first comprehensive textbook on BPM. He was awarded with the best paper award at the 11th International Conference on BPM. More information on Marcello can be found at www.marcellolarosa.com.

Arthur ter Hofstede is a Professor in the Information Systems School in the Science and

Engineering Faculty, Queensland University of Technology, Brisbane, Australia, and is Head of the Business Process Management Discipline. He is also a Professor in the Information Systems Group of the School of Industrial Engineering of Eindhoven University of Technology, Eindhoven, The Netherlands. His research interests are in the areas of business process automation and process mining.

Moe Wynn is a researcher in the field of Business Process Management (BPM) within the Information Systems School at Queensland University of Technology, Australia. She received her PhD in the area of business process automation/workflow management in 2007. Her current research interests include cost-aware BPM, risk-aware BPM, process automation, and process mining. She works on inter-disciplinary research projects (e.g., in the healthcare domain) and carries out collaborative research with industry partners. She has published over 45 refereed papers in international journals and conferences in the field of BPM in the past ten years. Her research appeared in the following journals: Information Sciences, Data and Knowledge Engineering, Information and Software technology, Formal Aspects of Computing, Journal of Computer and System Sciences, International Journal of Cooperative Information Systems, Transactions on Petri Nets and Other Models of Concurrency, Computers in Industry, and Journal of Information Technology Theory and Applications.