Processes Meet Big Data: Connecting Data Science with Process Science

Wil van der Aalst *Senior Member, IEEE,* and Ernesto Damiani *Senior Member, IEEE*

Abstract—As more and more companies are embracing Big data, it has become apparent that the ultimate challenge is to relate massive amounts of event data to processes that are highly dynamic. To unleash the value of event data, events need to be tightly connected to the control and management of operational processes. However, the primary focus of Big data technologies is currently on storage, processing, and rather simple analytical tasks. Big data initiatives rarely focus on the improvement of end-to-end processes. To address this mismatch, we advocate a better integration of *data science*, *data technology* and *process science*. Data science approaches tend to be process agonistic whereas process science approaches tend to be model-driven without considering the "evidence" hidden in the data. Process mining aims to bridge this gap. This paper discusses the interplay between data science and process science and relates process mining to Big data technologies, service orientation, and cloud computing.

Index Terms—Process Mining, Data Science, Process Science, Big Data, Service Orientation, and Cloud Computing.

1 Introduction

Big data is changing the way we do business, socialize, conduct research, and govern society [1], [2]. Big data has become a board-level topic and organizations are investing heavily in related technologies. At the same time it is not always clear how to derive value from data. Collecting large amounts of data does not necessarily lead to better processes and services. Moreover, analytics are often targeting particular tasks rather than the end-to-end process.

We would like to stress the importance of the process perspective in Big data initiatives. The aim of this paper is twofold.

1) We connect data science and process science by sketching the history of both disciplines. The data science discipline combines techniques from statistics, data mining, machine learning, databases, visualization, ethics, and high performance computing. Process science can be seen as the broader discipline covering process-centric approaches including Business Process Management (BPM), Workflow Management (WFM), Business Process Reengineering (BPR), Operations Research (OR), etc. We argue that one needs to carefully combine process-centric and data-centric approaches. This seems obvious, yet most data science (process science) approaches are process (data) agnostic. Process mining techniques aim to bridge this gap [3], [4].
2) Although process mining techniques have been around for a few years and tools are readily available (ProM, Disco, ARIS PPM, QPR, Celonis, SNP, minit, myInvenio, Perceptive, etc.), they are not a first-priority in most Big data projects. The focus of Big data technology projects (Apache Hadoop, Spark, Storm, Impala, etc.) is mostly handling huge volumes of highly diverse data while providing simple reporting and traditional analytics, targeting specific steps in the process. Consequently, there are few process mining efforts tailored towards Big data applications.

Therefore, we discuss the interplay between process mining and Big data technologies, service orientation, and cloud computing in this paper.

This paper is also intended as an introduction to our special issue of IEEE Transactions on Services Computing focusing on "Processes Meet Big Data". In a way, the contributions we selected for this special issue and the content of this paper can be seen as complementary: the former present novel techniques and technologies that promise to bridge the gap between Process and Data Science, while the latter focuses on open issues that still need to be solved for that bridge to become robust and easy to cross. In their paper "Multilevel Process Mining for Financial Audits", Michael Werner and Nick Gehrke present some sophisticated analytics techniques for process data and show how they can be used to detect hidden irregularities when auditing financial processes.

The work "Online Discovery of Declarative Process Models from Event Streams" by Andrea Burattin, Marta Cimitile, Fabrizio Maggi and Alessandro Sperduti deals with extracting models from high-bandwidth event streams. In their work "Event Correlation Analytics: Scaling Process Mining Using MapReduce-Aware Event Correlation Discovery Techniques", Hicham Reguieg, Boualem Benatallah, Hamid R. Motahari Nezhad and Farouk Toumani describe how the most widespread Big data computational paradigm (see also Section 5.1 of this paper) can be applied to efficiently compute correlation analytics. A different con-

- *W. van der Aalst is with the Department of Mathematics and Computer Science, Eindhoven University of Technology, PO Box 513, NL-5600 MB Eindhoven, The Netherlands. E-mail: see http://vdaalst.com*
- *E. Damiani is with Khalifa University/EBTIC, Abu Dhabi, UAE, on leave from the Department of Computer Science, Universitá degli Studi di Milano, Italy. E-mail: ernesto.damiani@kustar.ac.ae*

Manuscript received September 23rd, 2015.

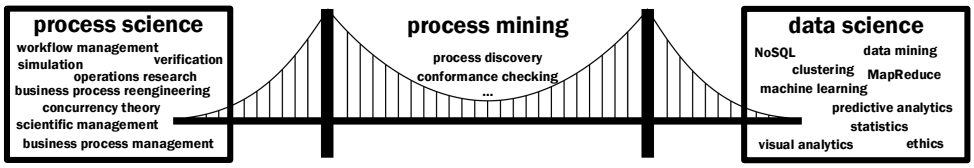

Fig. 1. Process mining as the bridge between process science and data science.

tribution to process discovery focusing on artefacts rather than on models is described in the paper "Discovering Interacting Artifacts from ERP Systems", by Xixi Lu, Marijn Nagelkerke, Dennis van de Wiel, and Dirk Fahland.

Other papers featured in this special issue focus more on process model synthesis, either by introducing data-intensive techniques for process model design, like the paper "An Automated Approach for Assisting the Design of Configurable Process Models" by Nour Assy, Nguyen Ngoc Chan and Walid Gaaloul, or by presenting bottom-up techniques to foster the emergence of processes via "spontaneous" service compositions, like "A Dataflow-Pattern-Based Recommendation Framework for Data Service Mashup" by Guiling Wang, Yanbo Han, Zhongmei Zhang and Shouli Zhang.

Finally, the papers "Analysis of Technology Trends Based on Diverse Data Sources" by Aviv Segev, Sukhwan Jung and Seungwoo Choi, and "Collaborative Agents for Distributed Load Management in Cloud Data Centers using Live Migration of Virtual Machines" by J. Octavio Gutierrez-Garcia and Adrian Ramirez-Nafar, discuss different aspects of the ongoing integration between process mining toolkits and Big data and Cloud Computing technologies (see also Section 5.2 of this paper).

In the remainder of this paper, we first review the history of process science (Section 2) and the history of data science (Section 3). Then, Section 4 introduces process mining as a means to bridge the gap between both. Section 5 discusses the implications and opportunities of implementing Process Mining tools on top of Big data technologies and platforms. First, we discuss the MapReduce programming model in the context of process mining (Section 5.1). Subsequently, we zoom in on the relationship between future Big-Data-enabled Process-Mining-as-a-Service, service orientation and cloud computing (Section 5.2). Section 6 concludes the paper.

2 A Brief History of Process Science

In recent years, "data science" has become a common term to refer to an emerging discipline revolving around the widespread availability of data. We would like to confront "data science" with the umbrella term "process science" which refers to the *broader discipline that combines knowledge from information technology and knowledge from management sciences to improve and run operational processes*. Process science aims at process improvements in terms of time, costs, quality, speed, flexibility, and reliability. This can be done through automation, but process science is definitely not limited to that. The process science discipline encompasses sub-disciplines such as Business Process Management (BPM). In this section, we provide a brief history of process science.

Since the industrial revolution, productivity has been increasing because of technical innovations, improvements in the organization of work, and the use of information technology. Adam Smith (1723-1790) showed the advantages of the division of labor. Frederick Taylor (1856-1915) introduced the initial principles of scientific management. Henry Ford (1863-1947) introduced the production line for the mass production of "black T-Fords".

Here we consider the birth of *scientific management* as the starting point of process science. Around 1880 Frederick Taylor, the "father of scientific management", got interested in answering questions like "What is the best way to do a job?". For example, Taylor found out that the optimal weight that a worker could lift using a shovel was 21 pounds [5]. This resulted in the "Wyoming 21-pound shovel" that exploited this scientific knowledge (see Figure 2).

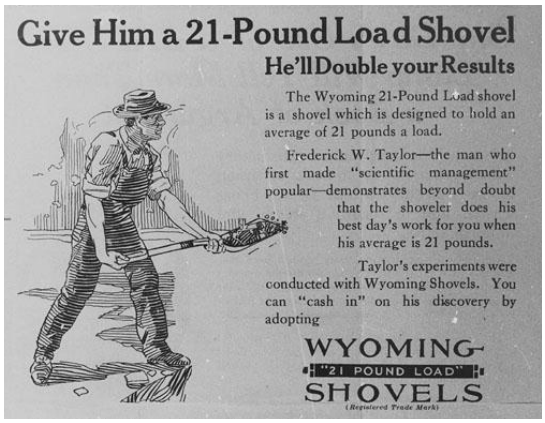

Fig. 2. Fragment of an advertisement by Wyoming Shovels from 1913 illustrating the origins of process science. Frederick Taylor empirically determined the optimal size of a shovel and this "scientific knowledge" was used to promote the "Wyoming 21-pound shovel" claiming that it could double or triple the productivity of workers.

Over time operational processes increased in complexity and the need to consider the "process as whole" rather than optimizing individual tasks has become more pressing. Rather than selecting the optimal size of a shovel, the need to orchestrate highly dynamic processes involving many entities emerged. The second industrial revolution addressed this need by realizing assembly lines and optimized processes (e.g., to produce "black T-Fords" as fast and cheap as possible). All of this was done without relying on sophisticated IT systems.

Around 1950, computers and digital communication infrastructures slowly started to influence business processes.

Two decades later the effects of this third industrial revolution became noticeable in every sector. The adoption of IT systems resulted in dramatic changes in the organization of work and enabled new ways of doing business. Today, innovations in computing and communication are still the main drivers behind change in almost all business processes. Business processes have become more complex, heavily rely on information systems, and may span multiple organizations. People are now using the term "Industry 4.0" [6] to refer to a fourth industrial revolution. The goal is to create "smart" manufacturing systems using a combination of embedded systems, sensors, networks, service orientation, Big data, and analytics.

Concurrent with the various stages of the industrial revolution, management science developed and started to use more sophisticated mathematical models to analyze and improve processes. *Operations Research* (OR) emerged as a scientific discipline [7]. OR employs a broad range of problem-solving techniques to improve decision-making and process performance, e.g., simulation, mathematical optimization, queueing theory, and Markov decision processes. Most of these techniques involve the construction of (mathematical) models that attempt to describe the underlying processes. Hence, OR is an essential ingredient of process science.

Over time, information systems (rather than people) started to "run" the operational processes. See, for example, the spectacular development that has brought SAP, a German corporation founded in 1972, to become the multinational world leader in enterprise software for managing business operations. Today, most large organizations use enterprise software from vendors like SAP, Oracle, IBM, HP, and Microsoft. Information systems manage and store data and these are used to *support and automate processes*.

Process descriptions (implicit or explicit) are used to create products and deliver services. As a result, process modeling has become of the utmost importance. Process models assist in managing complexity by providing insights and by documenting procedures. Information systems need to be configured and driven by precise instructions. Cross-organizational processes can only function properly if there is agreement on the required interactions. As a result, process models are widely used in today's organizations [8].

The importance of process models is also reflected by the uptake of *Workflow Management* (WFM) and *Business Process Management* (BPM) systems [8], [9], [10]. These are based on ideas already present in the early *office information systems*. Already in the seventies, people like Skip Ellis, Anatol Holt, and Michael Zisman worked on office information systems driven by explicit process models [8]. Ellis et al. developed office automation prototypes such as Officetalk-Zero and Officetalk-D at Xerox PARC in the late 1970s. These systems used variants of Petri nets to model processes. Another example from the same period is SCOOP (System for Computerizing of Office Processes), developed by Michael Zisman. SCOOP also used Petri nets to represent business processes. Officetalk, SCOOP and other office information systems were created in a time where workers were typically not connected to some network. Hence, these systems were not widely adopted. Nevertheless, the vision realized in today's BPM systems was already present.

In the mid-nineties there was the expectation that WFM systems would get a role comparable to Database Management (DBM) systems [9]. There was indeed a burst of WFM systems offered by a range of vendors. However, WFM systems were not widely adopted and did not manage to become and integral part of most information systems (like DBM systems). The early WFM systems were focusing too much on automation, not acknowledging the management aspects and the need for flexibility. Moreover, processes can also be captured in conventional programming languages. Hence, workflows are often hidden in code.

BPM can be seen as an extension of Workflow Management (WFM) [8]. WFM primarily focuses on the automation of business processes, whereas BPM has a broader scope: from process automation and process analysis to operations management and the organization of work. On the one hand, BPM aims to improve operational business processes, possibly without the use of new technologies. For example, by modeling a business process and analyzing it using simulation, management may get ideas on how to reduce costs while improving service levels. On the other hand, BPM is often associated with software to manage, control, and support operational processes.

Despite the availability of WFM/BPM systems, process management is not "subcontracted" to such systems at a scale comparable to DBM systems. The application of "pure" WFM/BPM systems is still limited to specific industries such as banking and insurance. *Process management turned out to be much more "thorny" than data management. BPM is multifaceted, complex, and difficult to demarcate.* Given the variety in requirements and close connection to business concerns, it is often impossible to use generic BPM/WFM solutions. Therefore, BPM functionality is often embedded in other systems. Moreover, BPM techniques are frequently used in concert with conventional information systems.

Process science, ranging from scientific management and operations research to WFM and BPM, is relevant for any organization. Considering the developments over the last century, productivity increased dramatically thanks to novel management principles, analysis techniques, and our ability to develop information systems that support and automate processes.

Process modeling and the analysis of process models both play an important role in process science. Hence, this brief history of process science would be incomplete without acknowledging this aspect [8]. Over time, many process modeling techniques have been proposed. In fact, the well-known Turing machine described by Alan Turing (1912-1954) can be viewed as a process model. It was instrumental in showing that many questions in computer science are undecidable. Moreover, it added a data component (the tape) to earlier transition systems. Also Markov chains, named after Andrey Markov (1856-1922), can be viewed as simple process models. However, Turing machines, transition systems, and Markov chains do not capture concurrency explicitly. Petri nets, proposed by Carl Adam Petri (1926-2010) in 1962, were the first formalism able to model concurrency. Petri nets play a prominent role in BPM: they are graphical and most of the contemporary BPM notations and systems use token-based semantics adopted from Petri nets. Concurrency is very important as in business processes many things may happen in parallel. Many cases may be

handled at the same time and resources may operate independently. Even within a case there may be various enabled or concurrently running activities. Therefore, systems and notations should support concurrency natively.

The analysis of process models should not be confused with the analysis of the process itself. Consider for example verification techniques (e.g., using model checking). These are more concerned with the internal consistency of the *model* (or software) rather than the performance of the *process*. Also simulation, queueing theory and Markov analysis only analyze models of the process rather than the process itself. Hence, the validity of the analysis results heavily depends on the faithfulness of the model. By exploiting event data generated by processes, the connection between model and reality can be ensured (cf. Section 4).

This concludes our brief, and very incomplete, history of process science. We skipped many developments and approaches in process science, e.g., Business Process Reengineering (BPR) [11], Six Sigma [12] and Case Management (CM) [13]. The many different ingredients illustrate the broadness and complexity of the discipline.

3 A Brief History of Data Science

Data science emerged as new discipline, simply because of the torrents of data that are collected *about anything*, *at any time*, and *at any place*. The overall volume of data recorded by mankind has been growing for decades [14], so in principle nothing is new. However, in many application domains it seems that a "tipping point" has been reached. Gartner uses the phrase "The Nexus of Forces" to refer to the convergence and mutual reinforcement of four interdependent trends: *social* (people want to connect, share, work and interact), *mobile* (people want do this at any location and at any time using multiple devices), *cloud* (people want to access any piece of data from anywhere), and *information* (people want to consume and generate information) [15]. This is directly impacting the way individuals, organizations, and societies interact, work, and do business [2]. Data science is not limited to "Big Data": also smaller data sets that easily fit on a single hard-drive may hold valuable information potentially changing the future of an organization. At the same time, also "small data" may be incredibly difficult to analyze (from both a computational and comprehension perspective).

Data science aims to use different data sources to answer questions grouped into the following four categories [1]:

- Reporting: *What happened?*
- Diagnosis: *Why did it happen?*
- Prediction: *What will happen?*
- Recommendation: *What is the best that can happen?*

Obviously, the above questions are highly generic illustrating the broadness of the data science discipline.

Over time, many alternative definitions of data science have been suggested [16]. In July 1966, Turing award winner Peter Naur proposed the term *datalogy* in a letter to the editor of the Communications of the ACM. He defined *datalogy* as the "science of the nature and use of data" and suggested to use this term rather than "computer science". Peter Naur also used the term "data science" long before it was in vogue. In [17], Naur writes: "A basic principle of *data science*, perhaps the most fundamental that may be formulated, can now be stated: The data representation must be chosen with due regard to the transformation to be achieved and the data processing tools available". The book from 1974 also has two parts considering "large data": "Part 5 - Processes with Large Amounts of Data" and "Part 6 - Large Data Systems". In the book, "large amounts of data" are all data sets that cannot be stored in working memory. The maximum capacity of magnetic disk stores considered in [17] ranges between 1.25 and 250 megabytes. Not only the disks are orders of magnitude smaller than today's disks, also the notion of what is "large/big" has changed dramatically since the early seventies. Some of the core principles of data processing may have remained invariant. However, in the early seventies it was impossible to imagine the amounts of data being collected today. Our ability to store and process data have developed in spectacular way as reflected by "Moore's Law" and all of its derivatives.

Data science is an amalgamation of different subdisciplines. Different authors will stress different ingredients. Here, we name a few:

- *Statistics*: The discipline is typically split into *descriptive* statistics (to summarize the sample data, e.g., mean, standard deviation, and frequency) and *inferential* statistics (using sample data to estimate characteristics of all data or to test a hypothesis). Clearly, data science has its roots in statistics.
- *Data mining*: In [18] data mining is defined as "the analysis of (often large) data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner". The input data is typically given as a table and the output may be rules, clusters, tree structures, graphs, equations, patterns, etc. Clearly, data mining builds on statistics, databases, and algorithms. Compared to statistics, the focus is on scalability and practical applications.
- *Machine learning*: The difference between data mining and machine learning is not a clear-cut. The field of machine learning emerged from within Artificial Intelligence (AI) with techniques such as neural networks. Here, we use the term machine learning to refer to algorithms that give computers the capability to learn without being explicitly programmed. To learn and adapt, a model is built from input data (rather than using fixed routines). The evolving model is used to make data-driven predictions or decisions.
- *Process mining*: Adds the process perspective to machine learning and data mining. Starting point is event data that are related to process models, e.g., models are discovered from event data or event data are replayed on models to analyze compliance and performance.
- *Stochastics*: Stochastic systems and processes behave in an unpredictable manner due to the influence of random variables. The stochastics discipline aims to estimate properties (e.g., flow time) in such uncertain contexts. Examples are Markov models and queue-

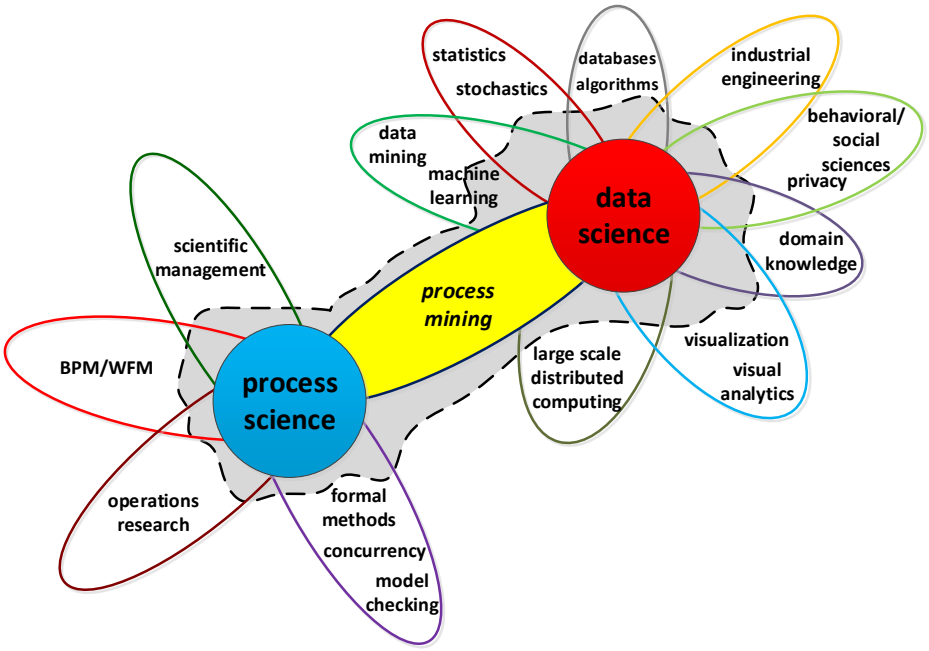

Fig. 3. Process mining as the linking pin between process science and data science.

ing networks whose parameters can be estimated from data.

- *Databases*: No data science without data. Hence, the database discipline forms one of the cornerstones of data science. Database Management (DBM) systems serve two purposes: (1) structuring data so that they can be managed easily and (2) providing scalability and reliable performance so that application programmers do not need to worry about data storage. Until recently, relational databases and SQL (Structured Query Language) were the norm. Due to the growing volume of data, massively distributed databases and so-called NoSQL databases emerged. Moreover, in-memory computing (cf. SAP HANA) can be used to answer questions in real-time.
- *Algorithms*: To handle large amounts of data and to answer complex questions, it is vital to provide algorithms that scale well. Algorithmic improvements allow for the analysis of problems that are orders of magnitude larger than before. Consider for example distributed algorithms based on the MapReduce paradigm or algorithms that provide approximate results in a fraction of time.
- *Large scale distributed computing*: Sometimes problems are too large to be solved using a single computer. Moreover, data need to be moved around from one location to another. This requires a powerful distributed infrastructure. Apache Hadoop is an example of open-source software framework tailored towards large scale distributed computing.
- *Visualization and visual analytics*: Data and process mining techniques can be used to extract knowledge from data. However, if there are many "unknown unknowns" (things we don't know we don't know), analysis heavily relies on human judgment and direct interaction with the data. The perception capabilities of the human cognitive system can be exploited by using the right visualizations [19]. Visual analytics, a term coined by Jim Thomas, combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex data sets [20].
- *Behavioral/social sciences*: Behavioral science is the systematic analysis and investigation of human behavior. Social sciences study the processes of a social system and the relationships among individuals within a society. To interpret the results of various types of analytics it is important to understand human behavior and the social context in which humans and organizations operate. Moreover, analysis results often trigger questions related to coaching and positively influencing people.
- *Industrial engineering*: Data science is often applied in a business context where processes should be as efficient and effective as possible. This may require knowledge of accounting, logistics, production, finance, procurement, sales, marketing, warehousing, and transportation.
- *Privacy and security*: Privacy refers to the ability to seclude sensitive information. Privacy partly overlaps with security which aims to ensure the confidentiality, integrity and availability of data. Data should be accurate and stored safely, not allowing for unauthorized access. Privacy and security need to be considered carefully in data science applications. Individuals need to be able to trust the way data is stored and transmitted.

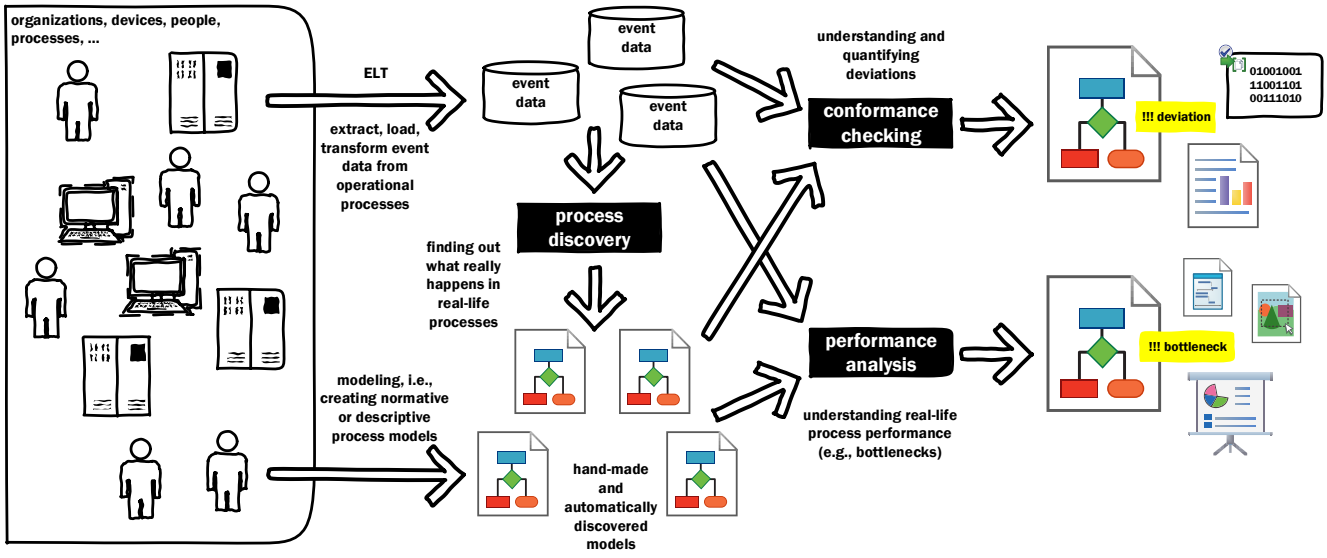

Fig. 4. Process mining overview showing three common types of process mining process discovery, conformance checking, and performance analysis.

- *Ethics*: Next to concrete privacy and security breaches there may be ethical notions related to "good" and "bad" conduct. Not all types of analysis possible are morally defendable. For example, mining techniques may favor particular groups (e.g., a decision tree may reveal that it is better to give insurance to white middle-aged males and not to elderly black females). Moreover, due to a lack of sufficient data, minority groups may be wrongly classified. Discrimination-aware data mining techniques address this problem by forcing the outcomes to be "fair" [21].

The above list nicely illustrates the interdisciplinary nature of data science.

4 Process Mining: Bridging the Gap

Process mining techniques can be used to extract process-related information from event logs [3]. They can be seen as part of the broader data science and process science disciplines. In fact, as Figure 3 shows, process mining is the linking pin between both disciplines. Process mining seeks the confrontation between event data (i.e., observed behavior) and process models (hand-made or discovered automatically). The interest in process mining is rising as is reflected by the growing numbers of publications, citations and commercial tools (Disco, ARIS PPM, QPR, Celonis, SNP, minit, myInvenio, Perceptive, etc.). In the academic world, ProM is the de-facto standard (www.processmining.org) and research groups all of the world have contributed to the hundreds of ProM plug-ins available.

The starting point for any process mining effort is a collection of *events* commonly referred to as an *event log* (although events can also be stored in a database). Each event corresponds to:

- a *case* (also called process instance), e.g., an order number, a patient id, or a business trip,
- an *activity*, e.g., "evaluate request" or "inform customer",
- a *timestamp*, e.g., "2015-09-23T06:38:50+00:00",
- additional (optional) *attributes* such as the *resource* executing the corresponding event, the *type* of event (e.g., start, complete, schedule, abort), the *location* of the event, or the *costs* of an event.

Event logs can be used for a wide variety of process mining techniques. Here we focus on the three main types of process mining: *process discovery*, *conformance checking*, and *performance analysis* (see Figure 4).

The first type of process mining is *discovery*. A discovery technique takes an event log and produces a process model (e.g., Petri net or BPMN model) without using any a priori information. Process discovery is the most prominent process-mining technique. Techniques ranging from the basic Alpha algorithm [3] to sophisticated inductive mining techniques [22] are available. For many organizations it is often surprising to see that these techniques are indeed able to discover real processes merely based on example behaviors stored in event logs. The discovered process models provide valuable insights, and provide the basis for other types of process mining (bottleneck analysis, deviation analysis, predictions, etc.).

The second type of process mining is *conformance checking* [3], [23]. 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 process model used as input may be hand-made or discovered. To check compliance often a normative hand-crafted model is used. However, to find exceptional cases, one often uses a discovered process model showing the mainstream behavior. It is also possible to "repair" process models based on event data.

State-of-the-art conformance checking techniques create so-called *alignments* between the observed event data and

the process model [23]. In an alignment each case is mapped onto the closest path in the model, also in case of deviations and non-determinism. As a result, event logs can always be replayed on process models. This holds for both discovered processes and hand-made models. The ability to replay event data on models and the availability of timestamps enables the third type of process mining: *performance analysis*. Replay may reveal *bottlenecks* in the process. Moreover, the tight coupling between model and data also helps to find root causes for performance problems. The combination of event data with process-centric visualizations enables novel forms of performance analyses that go far beyond traditional Business Intelligence (BI) tools.

5 Applying Big-Data Techniques to Process Mining

The growing availability of data over the last two decades has given rise to a number of successful technologies, ranging from data collection and storage infrastructures to hardware and software tools for analytics computation and efficient implementations of analytics. Recently, this technology ecosystem has undergone some radical change due to the advent of Big data techniques [24]. A major promise of Big data is enabling "full data" analysis [25]), i.e., the computation of analytics on all available data points, as opposed to doing so on selected data samples. In the case of process mining, "going full data" means that the process miner will be able to analyse huge amounts of event data, obtaining a complete picture of the process including all possible variations and exceptions.

5.1 MapReduce in the Context of Process Mining

Big data technology often relies on *MapReduce* [26], a basic computational paradigm, which has been remarkably successful in handling the heavy computational demands posed by huge data sets. MapReduce models a parallel computation as a sequence of rounds, each of which consists in computing a *Map* and a *Reduce* function over lists of *(key, value)* pairs. This paradigm is reminiscent of classic functional programming inasmuch access to input data is entirely *by value*: any change to data values that takes place on the computational nodes does not automatically propagate back to the original data entries. Communication occurs by generating new *(key, value)* lists which are then fed into the next round of execution.

Unlike what happens in classic functional programming, however, in MapReduce output lists need not have the same cardinality as the input ones. Rather, the Map function maps its input list into an arbitrary (but usually lower) number of values. Even more, the Reduce function usually turns a large list of pairs into one (or a few) output values. MapReduce performance gain comes from the fact that all of the Map output values are not reduced by the same Reduce node. Rather, there are multiple reducers, each of which receives a list of *(key, value)* pairs having the same key, and computes its Reduce function on it independently of reduce operations taking place at other nodes. The parallel operation of MapReduce is summarized in Figure 5.

In principle, the MapReduce paradigm looks well suited to implementing process mining algorithms. Nevertheless, some issues still stand in the way of MapReduce implementations of PM techniques. Indeed, MapReduce implementations of process mining algorithms like Alpha Miner, the Flexible Heuristics Miner (FHM), and the Inductive Miner [27] have been described only recently [28], [29].

Let us now focus on what needs to be done to use MapReduce in the context of process mining. Most open issues regard the Map function. MapReduce implementations of process mining algorithms should compute keys based on datatypes that are available within (or reachable from) process log entries, such as task, location and operator IDs or timestamps. In other words, the Map phase should put under the same key all process data that must be considered together when performing the final computation (the Reduce phase).

To understand why this is still an issue, let us consider a MapReduce program that computes the average (and the standard deviation) of a process' wall clock duration for different age intervals of the human operators who carry it out. This simple computation can be performed in a single round. Firstly, the program will map the list of *(operator_ID, process_duration)* pairs into a list of *(age_bucket_code, process_duration)*. Then, it will use the *(age_bucket_code)* to route *(key, value)* pairs to the Reduce nodes. Thirdly, each Reduce node will compute the average and standard deviation of its own *(key, value)* pairs. This computation is straightforward because the Map function is nothing else than a look-up table mapping employee IDs to age buckets. However, it is easy to see that if the process miner wanted to perform the same computation by skill/competence level rather than by age, computing the keys would become more complex, possibly involving inferences over equivalent competences. In case of complex mappings, the impact of the Map function on the total computation time can become very high, especially in the case of multiple rounds. Research is therefore needed to develop efficient Map functions for process mining, involving pre-computation of keys and even smaller scale parallelization.

Another open issue regards *load balancing*. The overall performance of MapReduce is highly dependent on carefully balancing the size of the lists to be handled at Reduce nodes. Such balancing can sometimes be achieved by writing *intelligent* Map functions. In our example, an intelligent Map function could consider the distribution of age among personnel, and send to the same Reduce node the age buckets that are known to hold few employees, while reserving an entire node for larger buckets. However, estimating (e.g., from process model analysis) or learning (e.g., by preliminary sampling event streams) the cardinality distribution of keys is by no means straightforward.

Some preliminary work has been done [30] on how to use domain taxonomies to improve the efficiency of MapReduce. Basically, if a set of events has been tagged using a taxonomy, a Map function can be written that generates keys corresponding to taxonomy subtrees of different depths, to generate lists of the same cardinality. For instance, let us assume that we want to tag process events that encode calls to a help desk by using a simple taxonomy of activities rooted in `Request`. Our taxonomy includes two subtrees, rooted respectively in `Repair` and `Replacement`. If we know (or can guess) that there will be nearly as many calls

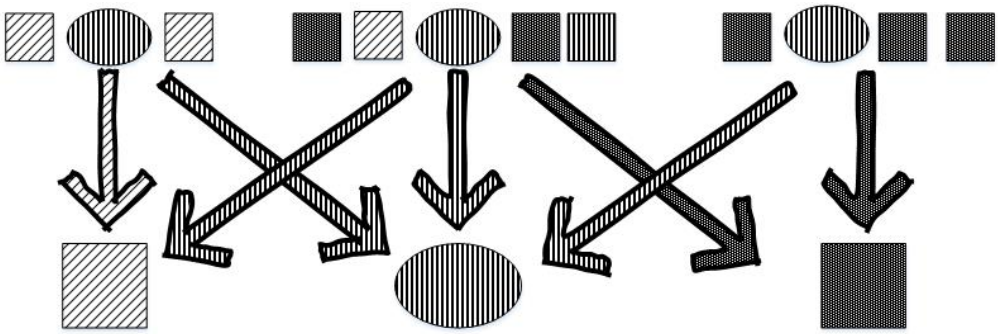

Fig. 5. The gist of MapReduce computational paradigm: all *(key, value)* pairs with the same key are sent to the same Reduce node

to be tagged with each child of `Replacement` as there are with `Repair`, we can use each child of `Replacement` (e.g. `BatterySubstitution`) to generate a key and `Repair` to generate another, balancing the cardinalities of *(key, value)* lists to be processed by MapReduce. However, estimating a priori the "right" depth of each subtree to be chosen for balancing the lists is not straightforward.

In the future, we hope to see tools for defining Map and Reduce functions on the basis of the business process model data types, of the relations among them and of other semantics-rich context information [31]. New techniques are also needed for exploiting semantics to balance load at the Reduce nodes. The process miner should be helped to write key-generation rules that automatically generate the same number of pairs for each key value, achieving balanced parallelization. Support for semantics-aware, intelligent Map functions is still very patchy in current Big data toolkits. Indeed, most of the initial Big data applications were targeted to unstructured processes (such as social media conversations), and many Big data environments do not yet support process models.

In terms of implementation, Map functions can be easily scripted to include simple conformance checks (Section 4) like spotting illegal event IDs. Unfortunately, scripting complex conformance checks may prove trickier. An initial attempt at automating checks within Big data tools was made by IBM within its *Business Insight Toolkit (BITKit)* [32]. Via its BigSQL utility, IBM Insight allows for defining a pseudo-SQL schema and to write pseudo-SQL queries against it that are automatically compiled for Hadoop as parallel computations. So, a business process miner could potentially spot a business process model violation simply by writing such a pseudo-SQL query. However, the expressive power of this approach is confined to the simple queries one can write in pseudo-SQL.

Recently, Big data technology has focused on building systems supporting a declarative, workflow-oriented approach to executing MapReduce computations. To mention but a few, these systems include Apache Oozie, Azkaban (originated at LinkedIn) and Luigi (developed at Spotify). These tools can be seen as potentially "BPM-friendly" in the sense of modelling Big data analytics itself as a workflow and allowing the process miner to insert user-defined modules in the data analysis pipeline. However, they do not yet include libraries for supporting standard Process Model checks, or other recurrent steps of BPM. Apache Flink, developed within the Stratosphere research project [33] looks promising for process mining applications. Flink combines stream processing and MapReduce computations, and supports inserting user defined functions, as well as complex data types.

Last but not least, a key requirement of process mining is automated support for *anonymization*, also known as *de-identification*, to protect privacy of personal data showing up in process logs [34] and preventing unauthorized uses of process-based metrics for worker discrimination. In the past decade, many computational notions of de-identification such as *k-anonymity*, *l-diversity*, and *t-closeness* have been introduced to customize privacy-preserving data mining to the requirements of specific domains [35]. However, unlike some BPM data preparation tools [36], Big data import toolkits do not yet support automated anonymization [37]. In particular, it is still unclear whether enforcement of privacy, trustworthiness and access control on process events is better done before mapping events to generate key values, or as part of the mapping, automatically bringing process *(key, value)* pairs to the granularity and detail level compatible with the privacy preferences and regulation compliance requirements of the process owners.

5.2 Cloud-Based Deployment for Process Mining

Conventional process mining tools are often deployed on the process owners' premises. However, cloud-based deployment seems a natural choice for Process-Mining-as-a-Service. Indeed, the elasticity of the cloud provisioning model [38] has suggested since long to deliver MapReduce on virtual clusters that can "scale out", requesting additional cloud resources when needed [39]. On the process analysis side, some process mining tools like Zeus [40] already included a scale-out capability on an internal grid architecture. Elastic provisioning of MapReduce clusters promises to add two key features to PM tools:

- Co-provisioning of data and analytics, i.e. run-time, computer-assisted choice of matching data representation and process mining algorithms according to user requirements.
- Negotiable Service Level Agreements (SLAs), in terms of delivery time and accuracy of the process mining results.

In the long term, performing process analysis on public clouds in co-tenancy with other process owners should facilitate cross-organizational process mining, encouraging organizations to learn from each other and improve their processes [41]. However, it is important to remark that deploying MapReduce on virtual (as opposed to physical) machines in a cloud infrastructure does not automatically exploit the elasticity that cloud computing can offer, because there is a non-trivial semantic gap that still needs to be bridged.

As we have discussed in Section 5.1, MapReduce parallelization is based on nodes that independently compute local instances of the Map and Reduce functions. However, today's cloud resource managers do not "see" these nodes; rather, they allocate computation units as Virtual Machines (VMs), which in turn are dynamically assigned to physical blades. As intuition suggests, lousy node-to-VM and VM-to-blade allocations could impair all the efforts made to achieve balanced key-to-node assignments in the MapReduce computation. In terms of the simple example we introduced in Section 5.1, what would be the point of a devising a balanced assignment of age buckets to Reduce nodes if these nodes would get randomly mapped to cloud VMs (and VMs to blades)?

Work is currently underway [42] on designing flexible MapReduce runtimes that seamlessly integrate semantics-aware Reduce functions with cloud resource allocation, but much remains to be done before this goal is fully reached.

6 Conclusion

Companies and organizations worldwide are increasingly aware of the potential competitive advantage they could get by timely and accurate "full data" process mining based on (1) sophisticated discovery and visualization techniques and (2) the BigData computational paradigm. However, as highlighted in this paper, several research and technology challenges remain to be solved before process mining, data science and Big data technologies can seamlessly work together.

The special issue "Process Analysis meets Big Data" of the IEEE Transactions on Services Computing features some notable contributions toward overcoming the hurdles that still prevent us from reaping the full benefits of Big data techniques in Process Mining. In this extended editorial paper, we have discussed the relation between process and data science, identified some of the remaining difficulties and outlined a research strategy that we believe should underlie the community's efforts toward full integration of Data and Process science. Hopefully, this vision will soon bring about scalable process mining analytics on top of Big data toolkits, while enabling them to be easily tailored to domain-specific requirements.

Further research is also needed to deliver the software environment for this to take place, possibly taking advantage of the available "architectural glue" to integrate process mining modules in the Big data pipeline. Hopefully, once the marriage between Big data and process mining has become a reality, process mining services will become more affordable, driving costs of process-aware analytics well within reach of organizations that do not have the expertise and budget for it today.

References

[1] W. van der Aalst, "Data Scientist: The Engineer of the Future," in *Proceedings of the I-ESA Conference*, ser. Enterprise Interoperability, K. Mertins, F. Benaben, R. Poler, and J. Bourrieres, Eds., vol. 7. Springer-Verlag, Berlin, 2014, pp. 13–28.

[2] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. Byers, "Big Data: The Next Frontier for Innovation, Competition, and Productivity," 2011, McKinsey Global Institute.

[3] W. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag, Berlin, 2011.

[4] W. van der Aalst, "Process Mining as the Superglue Between Data Science and Enterprise Computing," in *IEEE International Enterprise Distributed Object Computing Conference (EDOC 2014)*, M. Reichert, S. Rinderle-Ma, and G. Grossmann, Eds. IEEE Computer Society, 2014, pp. 1–1.

[5] F. Taylor, *The Principles of Scientific Management*. Harper and Bothers Publishers, New York, 1919.

[6] Bundesministerium für Bildung und Forschung, *Industrie 4.0: Innovationen für die Produktion von morgen*. BMBF, Bonn, Gemany, 2014, http://www.bmbf.de/pub/broschuere_Industrie-4.0-gesamt.pdf.

[7] J. Moder and S. Elmaghraby, *Handbook of Operations Research: Foundations and Fundamentals*. Van Nostrand Reinhold,New York, 1978.

[8] W. van der Aalst, "Business Process Management: A Comprehensive Survey," *ISRN Software Engineering*, pp. 1–37, 2013, doi:10.1155/2013/507984.

[9] W. van der Aalst and K. van Hee, *Workflow Management: Models, Methods, and Systems*. MIT press, Cambridge, MA, 2002.

[10] M. Dumas, M. Rosa, J. Mendling, and H. Reijers, *Fundamentals of Business Process Management*. Springer-Verlag, Berlin, 2013.

[11] M. Hammer and J. Champy, *Reengineering the corporation*. Nicolas Brealey Publishing, London, 1993.

[12] T. Pyzdek, *The Six Sigma Handbook: A Complete Guide for Green Belts, Black Belts, and Managers at All Levels*. McGraw Hill, New York, 2003.

[13] W. van der Aalst, M. Weske, and D. Grünbauer, "Case Handling: A New Paradigm for Business Process Support," *Data and Knowledge Engineering*, vol. 53, no. 2, pp. 129–162, 2005.

[14] M. Hilbert and P. Lopez, "The World's Technological Capacity to Store, Communicate, and Compute Information," *Science*, vol. 332, no. 6025, pp. 60–65, 2011.

[15] C. Howard, D. Plummer, Y. Genovese, J. Mann, D. Willis, and D. Smith, "The Nexus of Forces: Social, Mobile, Cloud and Information," 2012, http://www.gartner.com.

[16] G. Press, "A Very Short History of Data Science," Forbes Technology, http://www.forbes.com/sites/gilpress/2013/05/28/a-very-short-history-of-data-science/, 2013.

[17] P. Naur, *Concise Survey of Computer Methods*. Studentlitteratur Lund, Akademisk Forlag, Kobenhaven, 1974.

[18] D. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*. MIT press, Cambridge, MA, 2001.

[19] J. Wijk, "The Value of Visualization," in *Visualization 2005*. IEEE CS Press, 2005, pp. 79–86.

[20] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann, Eds., *Mastering the Information Age: Solving Problems with Visual Analytics*. VisMaster, http://www.vismaster.eu/book/, 2010.

[21] D. Pedreshi, S. Ruggieri, and F. Turini, "Discrimination-aware data mining," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2008, pp. 560–568.

[22] S. Leemans, D. Fahland, and W. van der Aalst, "Scalable Process Discovery with Guarantees," in *Enterprise, Business-Process and Information Systems Modeling (BPMDS 2015)*, ser. Lecture Notes in Business Information Processing, K. Gaaloul, R. Schmidt, S. Nurcan, S. Guerreiro, and Q. Ma, Eds., vol. 214. Springer-Verlag, Berlin, 2015, pp. 85–101.

[23] W. van der Aalst, A. Adriansyah, and B. van Dongen, "Replaying History on Process Models for Conformance Checking and Performance Analysis," *WIREs Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 182–192, 2012.

[24] O. Terzo, P. Ruiu, E. Bucci, and F. Xhafa, "Data as a service (daas) for sharing and processing of large data collections in the cloud," in *Complex, Intelligent, and Software Intensive Systems (CISIS), 2013 Seventh International Conference on*, July 2013, pp. 475–480.

[25] T. Pfeiffer, "Towards Distributed Intelligence Using Edge-Heavy Computing," 2015, http://ec.europa.eu/information_society/newsroom/cf/dae/document.cfm?action=display&doc_id=7682/.
[26] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.
[27] W. van der Aalst, T. Weijters, and L. Maruster, "Workflow mining: discovering process models from event logs," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 16, no. 9, pp. 1128–1142, Sept 2004.
[28] J. Evermann, "Scalable Process Discovery using Map-Reduce," *Services Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.
[29] S. Hernandez, S. Zelst, J. Ezpeleta, and W. van der Aalst, "Handling Big(ger) Logs: Connecting ProM 6 to Apache Hadoop," in *Proceedings of the BPM2015 Demo Session*, ser. CEUR Workshop Proceedings, vol. 1418. CEUR-WS.org, 2015, pp. 80–84.
[30] E. Jahani, M. J. Cafarella, and C. Ré, "Automatic Optimization for MapReduce Programs," *Proceedings of the VLDB Endowment*, vol. 4, no. 6, pp. 385–396, Mar. 2011.
[31] L. Bodenstaff, E. Damiani, P. Ceravolo, C. Fugazza, and K. Reed, "Representing and Validating Digital Business Processes," in *Web Information Systems and Technologies*, Lecture Notes in Business Information Processing, J. Filipe and J. Cordeiro, Eds. Springer Berlin Heidelberg, 2008, vol. 8, pp. 19–32.
[32] H. Ossher, R. Bellamy, D. Amid, A. Anaby-Tavor, M. Callery, M. Desmond, J. de Vries, A. Fisher, T. Frauenhofer, S. Krasikov, I. Simmonds, and C. Swart, "Business insight toolkit: Flexible pre-requirements modeling," in *Software Engineering - Companion Volume, 2009. ICSE-Companion 2009. 31st International Conference on*, May 2009, pp. 423–424.
[33] A. Alexandrov, R. Bergmann, S. Ewen, J.-C. Freytag, F. Hueske, A. Heise, O. Kao, M. Leich, U. Leser, V. Markl, F. Naumann, M. Peters, A. Rheinländer, M. J. Sax, S. Schelter, M. Höger, K. Tzoumas, and D. Warneke, "The Stratosphere platform for big data analytics," *The VLDB Journal*, vol. 23, no. 6, pp. 939–964, Dec. 2014.
[34] K. E. Emam, F. Dankar, R. Issa, E. Jonker, D. Amyot, E. Cogo, J.-P. Corriveau, M. Walker, S. Chowdhury, R. Vaillancourt, T. Roffey, and J. Bottomley, "A globally optimal k-anonymity method for the de-identification of health data," *Journal of the American Medical Informatics Association*, vol. 16, no. 5, 2009.
[35] A. Cavoukian and J. Jonas, "Privacy by Design in the Age of Big Data," Office of the Information and Privacy Commissioner, 2012, https://privacybydesign.ca/content/uploads/2012/06/pbd-big_data.pdf.
[36] C. Gunther and W. van der Aalst, "A Generic Import Framework For Process Event Logs," in *Proceedings of Business Process Management Workshops*, Sept. 2006, pp. 81–92.
[37] A. Cavoukian and D. Castro, "Big Data and Innovation, Setting the Record Straight: De-identification Does Work," Office of the Information and Privacy Commissioner, 2014, https://www.privacybydesign.ca/content/uploads/2014/06/pbd-de-identification_ITIF1.pdf.
[38] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010.
[39] C. Ardagna, E. Damiani, F. Frati, G. Montalbano, D. Rebeccani, and M. Ughetti, "A Competitive Scalability Approach for Cloud Architectures," in *Cloud Computing (CLOUD), 2014 IEEE 7th International Conference on*, June 2014, pp. 610–617.
[40] A. Azzini and P. Ceravolo, "Consistent Process Mining over Big Data Triple Stores," in *Big Data (BigData Congress), 2013 IEEE International Congress on*, June 2013, pp. 54–61.
[41] W. van der Aalst, "Configurable Services in the Cloud: Supporting Variability While Enabling Cross-Organizational Process Mining," in *On the Move to Meaningful Internet Systems: OTM 2010 - Confederated International Conferences*, Lecture Notes in Computer Science 6426, Springer 2010, pp. 8–25.
[42] D. Cheng, J. Rao, Y. Guo, and X. Zhou, "Improving MapReduce Performance in Heterogeneous Environments with Adaptive Task Tuning," in *Proceedings of the 15th International Middleware Conference*, ser. Middleware '14. New York, NY, USA: ACM, 2014, pp. 97–108.

Wil van der Aalst Prof.dr.ir. Wil van der Aalst is a full professor of Information Systems at the Technische Universiteit Eindhoven (TU/e). At TU/e he is the scientific director of the Data Science Center Eindhoven (DSC/e). Since 2003 he holds a part-time position at Queensland University of Technology (QUT) and is also a member of the Royal Netherlands Academy of Arts and Sciences, Royal Holland Society of Sciences and Humanities, and Academia Europaea. His personal research interests include workflow management, process mining, Petri nets, business process management, process modeling, and process analysis.

Ernesto Damiani Prof. Ernesto Damiani is currently a full professor and director of the Information Research Security Centre at Khalifa University, Abu Dhabi, UAE, where he also leads the Big data Initiative of the Etisalat British Telecom Innovation Centre (EBTIC). He is on leave from Universitá degli Studi di Milano, Italy, where he is the director of the SESAR Research Lab. Ernesto holds a visiting position at Tokyo Denki University, Japan and is a fellow of the Japanese Society for the Progress of Science. His research interests include Big data for Cyber-Security, process discovery and service-oriented computing.