Combining Semantic Lifting and Ad-hoc Contextual Analysis in a Data Loss Scenario

Antonia Azzini, Ernesto Damiani, Francesco Zavatarelli

Combining Semantic Lifting and ad-hoc Contextual Analysis in a Data Loss Scenario

Antonia Azzini, Ernesto Damiani, Francesco Zavatarelli

Università degli Studi di Milano, Dipartimento di Informatica
via Bramante - 65, 26013 Crema, Italy
{antonia.azzini,ernesto.damiani,francesco.zavatarelli}@unimi.it
http://www.springer.com/series/7911

Abstract. In this work we introduce the knowledge acquisition procedure supported in the KITE.it process management framework. We then illustrate how Process Mining technique are supported in this framework proposing a running example featured in a Data Loss scenario. We then provide some lesson learned that can be generalized in similar contexts. In particular, we show how applying an appropriate Semantic Lifting to the log may help to discover behavioral patterns of the process that is actually being executed. Our conclusions spotlight that it is viable to verify whether some non-functional properties hold during the process execution. Moreover, we describe the impact that Semantics Lifting has on support and confidence of the inferred probabilities of observing these behavioral patterns.

Key words: Business Process Monitoring, Semantic Lifting, Knowledge Acquisition Process

1 Introduction and Problem Description

Within Business Process Management (BPM), Business Process Intelligence (BPI) is a research area that is quickly gaining interest and importance: it refers to the application of various measurement and analysis techniques at design and at run-time in the area of business process management. In practice, BPI stands for an integrated set of tools that manage and evaluate process execution quality by offering several features such as monitoring, analysis, discovery, control, optimization and prediction.

In particular, within BPI, *process mining* (PM) is a process management technique that extracts knowledge from the events log recorded by an information system. It is often used for discovering processes when there is no *a priori* model, or for conformance analysis in case there is an *a priori* model to be compared with the events log in order to find out discrepancies between the log and the model.

BPI applications are typically focused on providing real-time access to data of business activity or business process, with specific attention to critical key performance indicators (KPIs), as also reported in the literature [1, 2, 3] with

the aim to improve the speed and the effectiveness of business operations. From a technological point of view, BPI requires the integration of operational business intelligence [4] with real-time applications.

A known limitation of BPM applications today is that, as demonstrated by [5], only a limited part of the auxiliary information of the business process execution is actually used.
According to the literature [6] the most important capabilities that are covered by BPM are the following.

- **Analysis and discovery**: supporting the analysis of process instances, from both business and IT perspective, in order to discover process models; this capability helps analysts to identify the causes for a particular process behavior.
- **Monitoring**: analyzing process under execution. This can either be in the sense of alerting the user that some unusual or unwanted situation is (about) to occur, or in the sense of interacting with the business process management system to prevent the violation of security and contractual policies as Service Level Agreements (SLA).
- **Optimization**: based upon information about past executions, identifying areas of possible improvement in business process definition and in resource allocation, so that, for example, bottlenecks or exceptional behaviors do not happen.
- **Prediction**: based on discovery capabilities and historic data examination, deriving prediction models and applying such models to running processes, to prevent the possibility of exceptions, undesired behaviors, and non-compliance.

However, the effort spent by the research community in developing these capabilities has not been equally allocated [5, 7]. Major part of the current efforts are devoted to the design of complete infrastructures for BPM. For instance, as reported in [7], several studies focused on designing and enacting models, discovering models from event logs, selecting models from collections, composing models and so on, while optimization and improvement were less considered. This lack of attention is probably rooted in the origins of BPM that distinguished itself from Database Management by taking a control-flow perspective on Information Systems [8]; but persists in the today literature.

For instance, Maturity Models (MM) [9] have been used very successfully in several areas, especially in software development, since they can provide assistance in improving code development and process testing and in handling the quality models. However MMs have been seldom, if ever, integrated with the results of BPM, impairing their ability to provide concrete guidelines on how to move form one maturity level to the next.

This work is rooted within the activities of the KITE.it project [10]. As described in Section 3 KITE.it gives particular attention to the metrics and their knowledge acquisition process. In fact, the KITE.it framework will support procedures such as *i)* creation, contextualization and execution of metrics, *ii)* connection between metrics and analysis paradigms, and *iii)* visualization of the

results. The final goal is to drive the monitoring process by following a logic of knowledge discovery, and derive previously unknown and potentially useful knowledge.

For this purpose, we introduced the KITE.it knowledge acquisition process that supports the evolution of analysis from descriptive to prescriptive and, finally, to predictive. In this work, we focus our attention on process mining techniques computing frequencies of event co-occurrencies to reconstruct the workflow of concurrent processes. In particular, we will show how applying an appropriate *semantic lifting* to the events and workflow log may help to discover the process that is actually being executed. In the Web scenario, the term semantic lifting refers to the process of associating content items with suitable semantic objects as metadata to turn unstructured content items into semantic knowledge. In our case, the semantic lifting procedure corresponds to all the transformations of low-level systems log carried out in order to achieve a conceptual description of business process instances, without having a priori knowledge of the business process.

In order to better illustrate our proposal, we present a case study based on a *data loss prevention scenario* that aim to prevent the loss of critical data. To describe our running example, we use a lightweight data and open standard representation model designed to support real time monitoring of business processes and based on a shared vocabulary called Resource Description Framework (RDF). We believe that the usage of RDF as modeling language allows independence and extremely flexible interoperability between applications.

The contributions of this paper are below summarized:

- An example on how semantic lifting may help to improve process discovering during process mining.
- A definition of a Data Loss Prevention System in RDF, modeling a multi-level security policy based on the organizational boundaries (internal vs external actors and resources).
- A "knowledge acquisition process" model, supporting the evolution of analysis form descriptive, to prescriptive, to inductive.
- Design of metrics on process instances, based on comparison with policies and KPIs defined at the strategic level, and exploitation of the inferences that can be generated from these comparisons in order to evolve metrics along the different steps of their life cycle;
- An example on how, using semantic lifting in combination with standard process mining techniques during the discovery phase, it is possible to extract not only knowledge about the structure of the process, but also verify if some non-functional properties, such as security properties, hold during the process execution.

This work is organized as follows. Section 2 proposes an overview on the literature focusing on process improvement, together with a brief introduction to the semantic lifting approach. Then, an introduction to the KITE.it methodology is provided in Section 3, giving also a short overview of the Resource Description

Framework. The metric definition workflow and the novel concept of "knowledge acquisition process" introduced in this paper are presented and discussed in Section 4, while in Section 5 we present our Data Loss scenario and we give some examples on how semantic lifting helps improving the investigation on the process. Concluding remarks are reported in Section 6.

2 Related Work

BPM and BPI emerged in the last years as a research program aimed at maximizing the value of the information available in a process. We claim that the full integration of metamodels (MMs) with BPM capabilities requires a new notion of metrics. In BPM, metrics are measures of quantitative assessment used for descriptive statistics, and comparison or for tracking process performances. Metrics are critical elements that capture the information carried out in a business process [11]. However, metrics are contextual in nature, and for this reason BPM frameworks often lack in providing procedures for driving their definition and life-cycle. A recent research involving about 700 process professionals, conducted by Forrester Research in conjunction with PEX Network, found inadequacies with current BPM metrics employed by many organization [12]. Most of the current studies on BPI focus on the analysis of the process behavior and result on performance improvement limited to this aspects [2].

We claim that a major role in this new notion of metrics will be played by semantic lifting techniques. The term semantic lifting refers to the process of associating content items with suitable metadata to turn unstructured content items into semantic knowledge resources. More specifically in [13], we claim that semantic lifting refers to all the transformations of low-level systems log carried out in order to achieve a conceptual description of business process instances. Typically, this procedure is carried out by converting data logs from the information system storage to a format suitable for process monitoring [14]. We claim that this problem is orthogonal to the abstraction problem in process mining, and deals with different levels of abstraction when comparing events with modeled business activities [15]: our goal is to see how it is possible to extract better knowledge about properties of the overall process when associating some semantics to a log event, and also how to map events and business activities/tasks.

So far, the term semantic lifting has been mainly used in the context of model-driven software development. In [16], the authors proposed a technique for designing and implementing tool components which can semantically lift model differences that arise among the different tools. In particular, they used the term semantic lifting of differences to refer to the transformation of low-level changes to all the more conceptual description of model modifications.

The literature [17] reports how BPM usually operates at two main distinct levels, that correspond, respectively, to a management level, supporting business organizations in optimizing their operational processes, and to a technology level, supporting IT users in process modeling and execution. In these two levels, experts operate without a systematic interaction and cooperation, causing the well

known problem of Business/IT disalignment. In order to reduce the gap between these two levels, in [17] De Nicola and colleagues refer to *s*emantic technologies as an useful approach for supporting the business process design, for reengineering, and maintaining the business process. The advantage regards the support that semantic lifting can give to business process design by semantically aligning a business process with a reference ontology. The semantic alignment can be achieved by performing consistency checking through the use of a reasoning engine. Then, business process reengineering can be improved by providing suggestions to experts during the design phase of the business process, for example finding alternative elements with semantic search and similarity reasoning over the business ontology. Another advantage is the possibility to support BP maintenance by automatically checking the alignment between one or more business processes against the business ontology when the latter is modified.

When mining a process, events need to be related to cases. As reported in the literature [18], this is a natural aspect, since a process model describes the life-cycle of a case of a particular type. In general, all mainstream process modeling notations specify a process as a collection of activities such that the life-cycle of a single instance is described.

Service-oriented business process development methodologies are usually based on a roadmap that comprises one preparatory phase to plan development, and several distinct phases that concentrate on business processes, like analysis and design (A&D), construction and testing, provisioning, deployment, execution and monitoring. The stepwise transition through these phases tends to be incremental and iterative in nature and should accommodate revisions in situations where the scope cannot be completely defined a priori. This allows the authors to describe an approach that is one of continuous invention, discovery, and implementation with each iteration, forcing the development team to drive the software development artifacts closer to completion in a predictable and repeatable manner [19]. Such a contribution considers multiple realization scenarios for business processes and Web Services that take into account both technical and business concerns.

Today BPI frameworks are mainly evolving in two directions:

- they no longer limit their analysis to process behavior but enlarge the scope to any other auxiliary data that is connected to process execution,
- they aim to exploit the acquired knowledge for predictive analysis.

Predictive analysis applied to process monitoring is often limited to, or strongly depended on temporal analysis. For instance, in [20], temporal logic rules are adopted to define business constraints, evaluated at execution time so that they can generate alerts that can prevent the development of violations. In [21], the authors present a set of approaches based on annotated transition systems containing time information extracted from event logs. The aim is again to check time conformance at execution time, as executions not aligned with annotated transitions predict the remaining processing time, and recommend countermeasures to the end users. In [22] a set of methods ranging from Markov

chains up to decision trees are applied and tested, underlining the characteristics that each approach exhibit.

Another approach for prediction of abnormal termination of business processes has been presented in [23]. Here, a fault detection algorithm (local outlier factor) is used to estimate the probability of abnormal termination. Alarms are provided to early notify probable abnormal terminations to prevent risks rather than merely reactive correction of risk eventualities. Other approaches go beyond temporal analysis extending predictive analysis to include ad-hoc contextual information. In [24], a clustering approach on SLA properties is coupled with behavioral analysis to discovered and model performance predictors. In [25], the authors propose an approach running statistical analysis on process-related data, notably the activities performed, their sequence, the resource availability, the capabilities and the interaction patterns. In [26], the authors propose an approach for Root Cause Analysis based on classification algorithms. After enriching a log with information like workload, occurrence of delay and involvement of resources, they use decision trees to identify the causes of overtime faults.

On the side of knowledge acquisition procedures some works are specifically oriented to the area of business process management [19]. However only a few are really considering analysis as a key element of this process. For instance, in [27] the authors exploit the notion of knowledge maintenance process. In this work, process mining is applied to analyze the knowledge maintenance logs to discover process and then construct a more appropriate knowledge maintenance process model. The proposed approach has been applied in the knowledge management system.

Our own work [28] is characterized by the introduction of an extended notion of process behavior that provides a systematic approach to capture process features beyond workflow execution. This element is then exploited within a knowledge acquisition methodology that uses prescriptive and predictive analysis to acquire novel and unexpected knowledge.

3 KITE Methodology and RDF Graphs

KITE.it is a project co-funded by the Italian Ministry for Economic Development, within the "Industria 2015" Programme, in the area of "New technologies for Made in Italy". KITE.it is aimed at developing a methodological and technological framework to support the evolution of the aerospace Supply Chains (SP) towards Value Network (VN) models. More specifically, the goals of the project are:

- supporting interoperability and cooperation among business networks and knowledge workers,
- monitoring the intellectual capital created and exchanged in the network, and
- measuring and optimizing performances.

In order to achieve these objectives, the project provides an environment for business and social cooperation that enables interoperability and cooperation

among enterprises and knowledge workers, making available all methodological and technological tools developed in the project. The KITE.it environment includes:

- *A general methodology (KITE.it Methodology)* to support the activities of analysis, design and monitoring, needed for the transition towards VN models. This methodology, shown in Figure 1, manages iteratively the entire business life cycle both at strategic and operational level. At the strategic level the exogenous variables and the VN, in which the organization operates, are analyzed. At the operational level, the strategy and the corporate policies are realized by an architecture of core processes.
- *A design environment*, that includes tools and languages for process and IT services design. The system includes an integrated metamodel for the multidimensional business process modeling and a modeler that designs the different business models or diagrams, for the description of the process architecture, the organizational structure, processes at various levels of detail, the business policies and the operational risks associated with the processes, as reported in Figure 2. In particular the integrated metamodel includes a model for metrics management.
- *A Business Performance Evaluation Environment (BPEE);* as shown in Figure 3, the system allows the definition of strategic and operative metrics, metrics on policies and risks (violations and dysfunctional behavior), and metrics about security and Social Network Analysis (SNA). A dashboard can display all these metrics. The system allows to monitor and optimize the performances of the value network and improve the knowledge on the business process or on the business model.

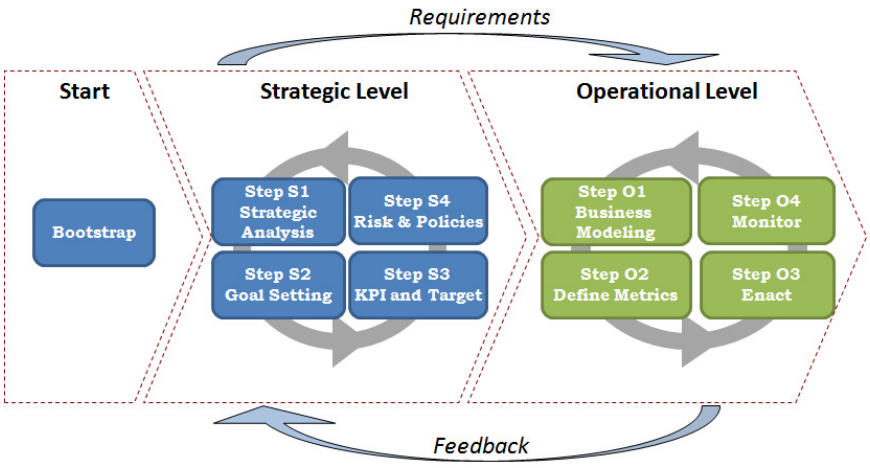

Fig. 1. A diagram illustrating the Kite Methodology.

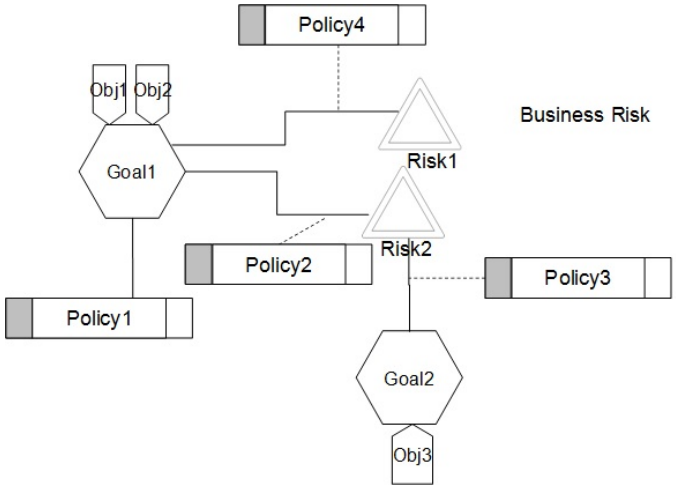

Fig. 2. The KITE.it design environment that allows the definition of business policies and operational risks associated with the processes.

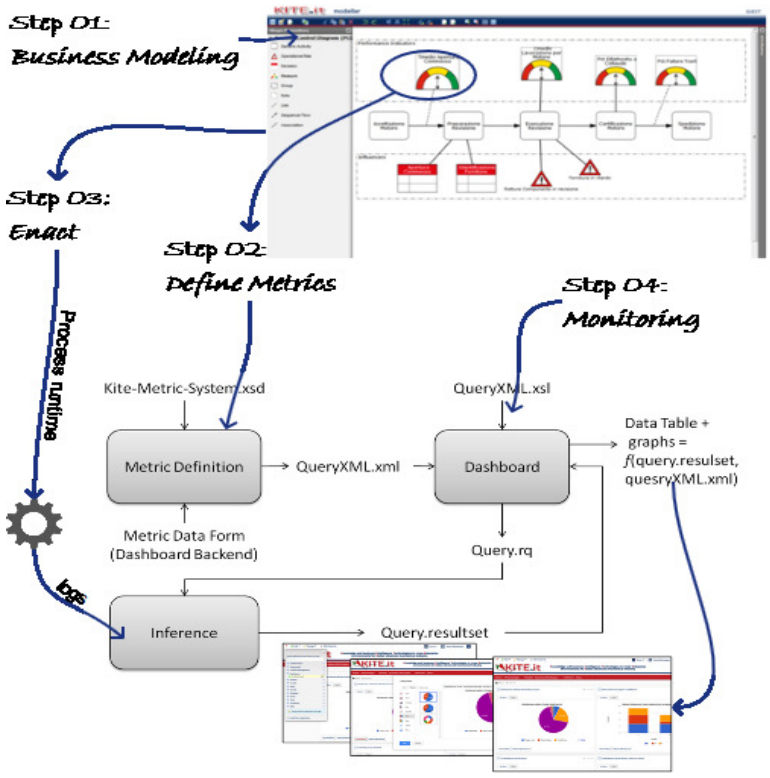

Fig. 3. The Business Performance Evaluation Environment (BPEE).

The goal is to drive the monitoring process by following a logic of knowledge discovery composed of four steps: (i) model the knowledge; (ii) acquire the data (iii) generate the inference; and (iv) derive previously unknown and potentially useful knowledge. To do this, KITE.it framework provides a set of tools and services that map all the monitoring assets.

In Section 3.1 we introduce the data representation model adopted by the project. In Section 4.1 we illustrate the metric model that defines the measurement procedures; consequently, in Section 4.2 we present the knowledge acquisition process implemented in KITE.it, and illustrate how this knowledge acquisition is leveraged through metric evaluation in the process iterations.

3.1 RDF Graphs

The BPEE has to integrate a variety of heterogeneous data from the different sources that compose KITE.it Environment. This requirement is fulfilled by adopting a monotonic data structure that supports the acquisition of new information without invalidating previously acquired knowledge. In particular, we adopted the so-called Resource Description Framework (RDF). Generally speaking, the RDF [29] corresponds to a standard vocabulary definition, which was proposed as the basis of the Semantic Web vision, and is composed by three elements: concepts, relations between concepts and attributes of concepts. These elements are modeled as a labelled oriented graph [30], defined by a set of triples $< s, p, o >$ where s is subject, p is predicate and o is object.

New information is inserted into a RDF graph by adding new triples to the set. It is therefore easy to understand why such a representation can provide big benefits to real time business process analysis: data can be appended 'on the fly' to the existing one, and it will become part of the graph, available for any analytical application, without any need to reconfigure or any other data preparation steps. RDF graphs allow external applications to query data through a standard language called SPARQL [31]. SPARQL is based on conjunctive queries on triple patterns, and allows to identify paths in the RDF graphs. Thus, queries can be seen as graph views. SPARQL is supported by most of the triples stores available. For all these reasons, RDF is an extremely generic data representation model that can be used in any domain.

In [32], the authors present a RDF-based framework for business process monitoring and analysis. They define a RDF model that represents a very generic business process and that can be easily extended to describe any specific process just by extending the RDF vocabulary. The model can be used as a reference for both monitoring applications (i.e., applications that produce data to be analyzed) and analysis tools. On one side, a process monitor creates and maintains the extension of the generic business process vocabulary either at start time, if the process is known a priori, or at runtime while capturing process execution data, if the process is not known. Process execution data is continuously saved as RDF triples that respect the extended model. On the other side, the analysis tools send SPARQL queries to the process under execution.

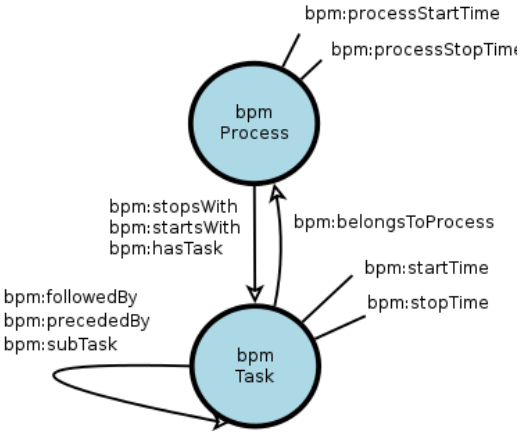

Fig. 4. RDF Representation of a generic business process.

Figure 4 shows the conceptual model of a generic business process, seen as a sequence of different tasks, each of them having a start/end time.

4 KITE Metric System

4.1 Metric Definition Workflow

KITE.it metric definition process allows to define metrics at strategic and operational level; the methodology was shown in Figure 1. All the metrics are monitor- and check- oriented: a general process is defined at strategic level through the goals definition, and it is then validated over the actual data. A crucial step of such a process is the definition of targets and KPIs [33], as well as risks and policies defined at strategic level. The metric definition step refers to a group of meta model specifications aiming at the implementation of the final metrics. It includes metric descriptors that have been modeled according to the XML Schema defined in KITE.it metric meta model. This metamodel is divided into four main areas (Figure 5), i.e. measurement, observation, computation and visualization, whose roles are described below.

- *Measurement*: it describes the network with strategic objectives (targets and policies); this refers to the analysis paradigm for the objectives description.
- *Observation*: it describes the metric observation conditions in terms of process instances or validity time of the data that will be analyzed. Il also allows to define thresholds or validity time for critical metric values, like those indicating the success in an objective achievement.
- *Computation*: it represents the operations that have to be carried out over the elements analyzed by the metric. It also describes the data access procedures, in terms of localization of the data sources required for the metric computation.

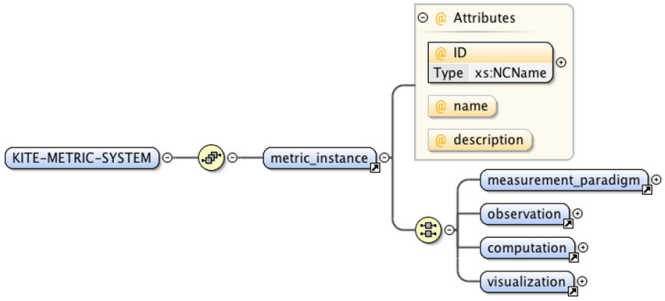

Fig. 5. Main components of KITE.it metric metamodel (KMM).

– *Visualization*: it represents the structures that will be used for metric data visualization.

The components of each of the four KITE.it metric metamodel (KMM) areas are shown in Figure 6.

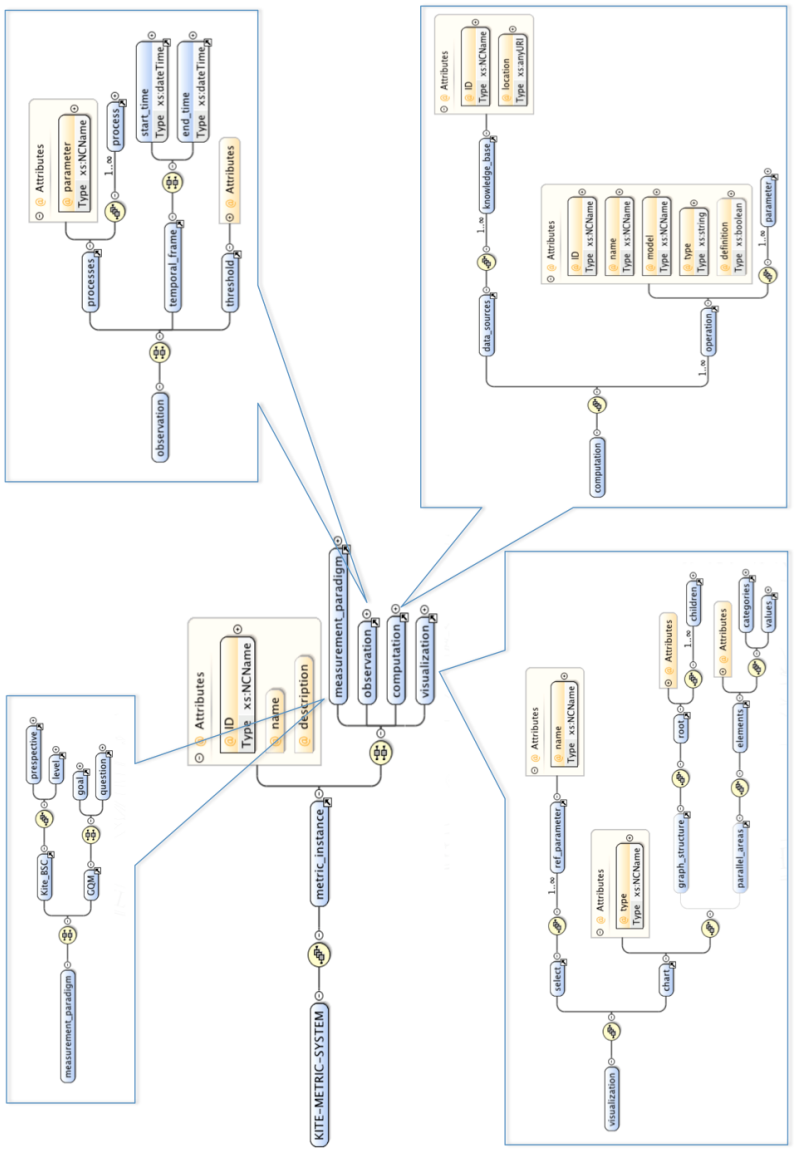

Fig. 6. Major details of the KITE.it metric metamodel (KMM).

4.2 Knowledge Acquisition Process

As previously reported in Section 1, KITE.it introduced a process that describes how acquisition of knowledge works and that is activated by the evaluation of the metrics adopted for the process execution monitoring.

The metrics considered in our approach can be classified into three main categories:

- *Descriptive*: they describe the process data and the behavioral attributes.
- *Prescriptive*: they evaluate the objective achievements, as well as KPIs and policy violations.
- *Inductive*: they are aimed at acquiring novel knowledge on the process through the evaluation of the statistic significance of the assumptions.

The first two categories of metrics are generally used to perform correlation analysis that then can generate the third category: the inductive metrics.

Our knowledge acquisition process organizes metrics along three development steps. Figure 7 shows a diagram of such a process.The main idea behind the knowledge acquisition process is the definition of "check points" that allow a better understanding of the role that each metric can play within a BPM process, and provide feedback to the monitoring process. Our "check-points" correspond to measures that support the evolution of a metric. In this work we identify two types of "check-points". The first type of check point involves the evolution from descriptive metrics into a prescription norm. Any metric can be considered descriptive, while we have a prescription when we assess the achievement of an goal or the deviation from a standard process behavior.

First of all, the descriptive metrics, applied to KITE.it knowledge base, defines the data that, together with the criteria considered to define the prescriptive metrics, are checked by applying the first "check-point". The result corresponds to the prescription norm. Several criteria can be adopted to define a prescription: confidence, support, maximum separation, etc. In this case we adopt confidence and maximum separation.

Then, a different descriptive metric is applied over the data deriving from the prescription and the knowledge base partitions. The resulting information is then used as input to the second type of "check-point", with other process data and the second set of criteria, in order to define a novel inductive analysis. At this point the iterations restart from the first "check-point", applied over new inductive analysis and data.

An example can be given by handling the quality levels obtained by maintenance technicians in a particular maintenance operation. Table 1 shows, respectively, the number of technicians that were evaluated by customers and the corresponding grade obtained in the operation. In this example a descriptive metric that reports the grades given by customers to maintenance technicians can be considered prescriptive when specific values are identified for discriminating among technicians following a regular process or not.

These values can be defined a priori based on an objective to be achieved, or a posteriori based on a data analysis. In the first case one could set the

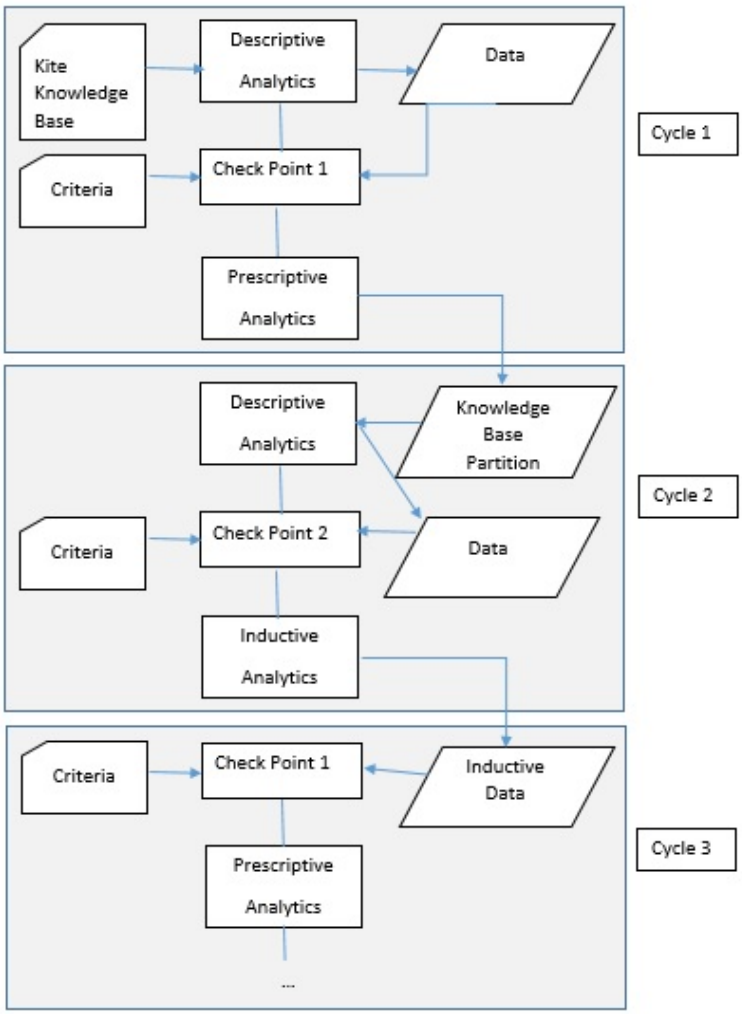

Fig. 7. Diagram of our proposed knowledge acquisition process.

grade C as the minimum value to consider a technician's performance. In the second case one could apply a maximum separation norm to identify $B - C$ as the area of maximum separation among the instances considered in Table 1. As the difference between the number of technicians A and B corresponds to 3, the difference between B and C is 7, and the difference between C and D is 1, we derive that the point of maximum separation is between B and C. We can therefore consider a prescription norm to be the situation in which a technician is able to achieve A or B for this performance to be considered regular.

The second "check-point" arises from the combination of descriptive analysis and prescription norm. We start by separating process instances into two sets:

Table 1. Example of separation norm

# of Technicians	Grades given by customers
13	A
10	B
3	C
4	D

those which are violating and those which are not violating a prescription norm. Then, we compare these two sets to verify if they exhibit significant differences on some properties. For instance, a correlation could be identified between the technicians violating the prescription (i.e. having grade differences lower than B) and technicians not performing specific operations.

The following definition allows to clarify how our knowledge acquisition process checks the different evolutions steps: a metric that is defined to monitor descriptive policies can be considered prescriptive if it allow to detect violations. Then descriptions and prescriptive metrics can be used to derive inductive knowledge that can be validated by data analysis.

5 Data Loss Case Study

Protecting the confidentiality of data stored in a information system or transmitted over a public network is a relevant problem in computer security. Generally speaking, data loss can be defined as an error condition in information systems in which data is destroyed by failures or negligence in storage, transmission, or processing operations. Consider for example companies that belong to the same manufacturing supply chain and that share business process critical data by using a common file server. This scenario could expose critical data to malicious users if access control is not implemented correctly. Indeed, access control to such file server should be carried out according to a security model, based on specific rules that consider, for example, the user authentication in file sharing activities, the definition of security policies for users that have access rights to some confidential data, a strictly defined control check before any operation of data sending and so on. In order to prevent data loss, systems usually develop an intellectual ownership defense scheme, also called *data-loss model* [34], which tracks any action operated on a document.

In our case study we consider two security levels generated by the organizational boundaries that define an user as internal or external.

As known from the literature [35], process mining is the technique of distilling a structured process description from a set of real executions. Here, we limit our example to process mining algorithms that are based on detecting order relations among the events that characterize a workflow execution log [36]. In particular, we build dependency/frequencies tables that are used to compare single executions in order to induce a reference model, or to verify the satisfia-

bility of specific conditions on the order of executions of events. The following definitions correspond to basic notions in this scenario [37].

Workflow trace. Let $E = \{e_1, e_2, \cdots, e_n\}$ be a set of events, then $t \in E^*$ is a *workflow (execution) trace.*

Workflow log. Let $E = \{e_1, e_2, \cdots, e_n\}$ be a set of events, then $W \subseteq E^*$ is a *workflow log.*

Successor. Let W be a workflow log over E and $a, b \in E$ be two events, then b is a *successor* of a (notation $a \prec_W b$) if and only if there is a trace $t \in W$ such that $t = \{e_1, e_2, \cdots, e_n\}$ with $e_i \equiv a$ and $e_{i+1} \equiv b$. Similarly, we use the notation $a \prec_W^n b$ to express that event b is *successor* of event a **by** n **steps** (i.e., $e_i \equiv a$ and $e_{i+k} \equiv b$, with $1 < k \leq n$).

The successor relationship is rich enough to reveal many workflow properties since we can construct dependency/frequency tables that allow to verify the relations that constraints a set of log traces. However, in order to better characterize the significance of dependency between events, also other measures, are adopted in the literature, such as for instance the J-Measure proposed by Smyth and Goodman [38], able to quantify the information content of a rule.

Table 2 shows a fragment of a workflow log generated by a data loss prevention system that tracks in-use actions. It is based on the RDF model described in the previous section. The system reports all the events that generate a new status of a specific document. In particular, we assume that for each event it is specified: (*i*) the type of event (Create, Update, Share, Remove); (*ii*) the user, who performed the action on the file, expressed by the email address, (*iii*) the event timestamp (this information allows to chronologically order all the events), and (*iv*) the status of the document.

5.1 The Data Loss Knowledge Acquisition Process

We now can construct the dependency/frequency (D/F) table [36] from the data log illustrated in Table 2. Table 3 shows descriptive metrics about D/F, in particular the metrics reported are:

- the overall frequency of event a (notation $\#a$);
- the frequency of event a followed by event Create (C for short);
- the frequency of event a followed by event Update (U for short) by 1, 2 and 3 steps;
- the frequency of event a followed by event Share (S for short) by 1, 2 and 3 steps.

Using data in Table 3 we can identify the process metrics divided into three main categories (descriptive, prescriptive, inductive) as introduced in previous Section. In particular, the Table summarizes the result of a descriptive analytic extracting the frequencies of the successor relation between events. Using this data we can now derive prescriptions through the distances reported between an event and the following one. Then, the expected behavior in the form of unwanted behaviors (black-listing) or wanted behavior (white-listing) needs to be defined. This can be done by identifying behavioral patterns in the workflow

Table 2. An example of workflow log for the Data Loss case study.

rec	Event	User	Timestamp	Status
File AAAA				
1	Create	userP@staff.org	2012-11-09 T 11:20	Draft
2	Update	userP@staff.org	2012-11-09 T 19:20	Draft
3	Share	userA@staff.org	2012-11-12 T 10:23	Proposal
4	Update	userA@staff.org	2012-11-14 T 18:47	Proposal
5	Share	userP@staff.org	2012-11-15 T 12:07	Proposal
6	Update	userP@staff.org	2012-11-18 T 09:21	Recommendation
7	Share	userM@inc.org	2012-11-18 T 14:31	Recommendation
File AAAB				
8	Create	userF@staff.org	2012-12-03 T 09:22	Draft
9	Update	userF@staff.org	2012-12-03 T 12:02	Draft
10	Update	userF@staff.org	2012-12-03 T 17:34	Draft
11	Share	userV@staff.org	2012-12-05 T 11:41	Draft
12	Share	userD@staff.org	2012-12-05 T 11:41	Proposal
13	Update	userD@staff.org	2012-12-08 T 10:36	Proposal
14	Update	userV@staff.org	2012-12-08 T 16:29	Proposal
15	Share	userG@inc.org	2012-12-10 T 08:09	Proposal
16	Update	userV@staff.org	2012-12-10 T 18:38	Recommendation
File AAAC				
17	Create	userA@staff.org	2012-12-04 T 10:26	Draft
18	Update	userA@staff.org	2012-12-04 T 13:12	Draft
19	Update	userA@staff.org	2012-12-05 T 10:12	Draft
20	Share	userV@staff.org	2012-12-05 T 12:22	Draft
21	Share	userD@staff.org	2012-12-06 T 14:51	Proposal
22	Share	userM@inc.org	2012-12-07 T 10:31	Proposal
File AAAD				
23	Create	userD@staff.org	2012-12-07 T 16.15	Draft
24	Share	userB@inc.org	2012-12-12 T 10:31	Proposal
25	Share	userM@inc.org	2012-12-12 T 12.37	Proposal
26	Share	userD@staff.org	2012-12-15 T 09.35	Proposal

Table 3. An example of Dependency/Frequency based on the Successor relation.

a	$\#a$	$a \prec$ C	$a \prec$ U	$a \prec^2$ U	$a \prec^3$ U	$a \prec$ S	$a \prec^2$ S	$a \prec^3$ S
Create	4	0	3	2	1	1	2	3
Update	10	0	3	3	2	6	5	5
Share	12	0	4	2	2	5	4	1

logs. Small values of dependency/frequency ratio, as well as high distances, may indicate anomalies in the process behavior.

In our case study a prescription defining constrains to be not violated can be expressed by the following expressions:

$$\begin{array}{c} Create \prec Share \\ Create \prec^2 Share \end{array} \quad (1)$$

The constraint such as 1 can be decomposed in two constituents: a `relation` and a `value`. In our case we have: `relation`: $\{\prec, \prec^2\}$, and `value`: {`Share`}.

An inductive analysis is then derived in two steps: in the first step, using prescription 1, we can partition the process in two sets, isolating process instances violating or not violating 1. Table 4 lists the records identified by the `relation` constituting constraint 1. Traces violating the prescriptions are *File AAAA* and *File AAAD*, by complement *File AAAB* and *File AAAC* are not. Events violating the prescriptions are at record 3, 24 and 25. In the following we are referring to these two partitions by using the notations Pv and $\overline{Pv}$. In particular we use the P^tv and $\overline{P^tv}$ to refer to set of traces and P^ev and $\overline{P^ev}$ to refer to set of events.

Table 4. An example of workflow log after contextualization of violation constraints.

rec	Event	User	Timestamp	Status
File AAAA				
2	Update	userP@staff.org	2012-11-09 T 19:20	Draft
3	Share	userA@staff.org	2012-11-12 T 10:23	Proposal
File AAAB				
9	Update	userF@staff.org	2012-12-03 T 12:02	Draft
10	Update	userF@staff.org	2012-12-03 T 17:34	Draft
File AAAC				
18	Update	userA@staff.org	2012-12-04 T 13:12	Draft
19	Update	userA@staff.org	2012-12-05 T 10:12	Draft
File AAAD				
24	Share	userB@inc.org	2012-12-12 T 10:31	Proposal
25	Share	userM@inc.org	2012-12-12 T 12.37	Proposal

The second step involves the analysis of the incidence of other process attributes on the partitions Pv and $\overline{Pv}$.

Table 5 shows the incidence of a descriptive analysis with respect to the set of process users on the sets P^tv and $\overline{P^tv}$. In this case the descriptive analysis is insisting on a single attribute, `User`, but in principle it could involve any projection over the dataset.

To quantify the incidence of a descriptive analysis to a specific partition we adopt Bayesian inference, as expressed in equation 2, where P is a partition over the data set and P_d is the set resulted from the projection of a descriptive analysis d. The "A-Priori" probability of observing a trace in P_v or in $\overline{P_v}$ given a trace with a specific value of `User`, is reported in Table 6. This probability

Table 5. Violation/Non-Violation Prescriptive Data Correlation.

Prescriptive Data Relation	UserA	UserB	UserD	UserF	UserG	UserM	UserP	UserV	Total
Violation (P_v)	2	1	0	0	0	1	0	0	3
Non Violation ($\overline{P_v}$)	3	0	5	3	1	2	4	4	9
Total	5	1	5	3	1	3	4	4	12

value expresses the probability of finding a violation or a non violation knowing the value of a descriptor d, for a specific data set. The column Δ proposes the difference between the two probabilities, allowing to quantify how likely an attribute can be associated to a partition of the data set.

$$\Pi(P|P_d) = \frac{\Pi(P) \cdot \Pi(P_d|P)}{\Pi(P_d)} \tag{2}$$

Table 6. "A-Priori" probability of observing a specific value of `User` in $P^t v$ or in $\overline{P_v^t}$.

User	$\Pi(P_v\|User)$	$\Pi(\overline{P_v}\|User)$	Δ
A	0.4	0.6	-0.2
B	1	0	1
D	0	1	-1
F	0	1	-1
G	0	1	-1
M	0.33	0.66	-0.33
P	0	1	-1
V	0	1	-1

Similarly, we can quantify the incidence of the attribute `User` on the data set contextualization defined by the relation $\{\prec, \prec^2\}$ (Table 4) and filtering the records by event, as reported in Table 7.

Table 7. "A-Priori" probability of observing a specific value of `User`, in $P^e v$ or in $\overline{P_v^e}$.

User	$\Pi(P_v\|User)$	$\Pi(\overline{P_v}\|User)$	Δ
P	0	1	-1
A	0.5	0.5	0
F	0	1	-1
V	0	1	-1
M	1	0	1
B	1	0	1

5.2 The Semantic Lifting

Using the notions previously introduced, we are now illustrating the impact that Semantic Lifting can have on the knowledge acquisition process. Semantic Lifting refers to the process of associating descriptors in a workflow log to implicit semantics through definitions that are not available in the structure of the workflow log itself. As any definitional operation, it has impact on the extensions of the instances belonging to the definition. In other words a Semantic Lifting procedure can extend or restrict the set of elements to be considered in our incidence analysis.

Back to our example, we can observe the behavior of the documents shared inside and outside the organization boundaries. Focusing our attention on the events restricted by constraint 1, we can see which traces might cause unwanted information flows. To this aim, a semantic lifting procedure is applied to the log data for remodeling the representation of the process and allowing additional investigations. In our example, the lifting can be made by introducing two new concepts: the `Internal` and the `External` users.

Figure 8 shows how values in the file `User` of our log are mapped on the new concepts. The two data transformations rules expressed in Equation 3 implement this mapping, which are integrated in our data model using standard techniques for RDF data [39].

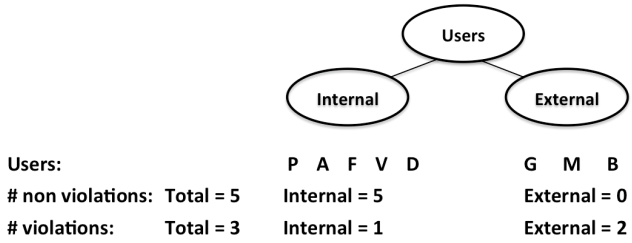

Fig. 8. Internal and external violations subsets.

$$\begin{gathered} User \| \left[A-Z0-9\_.-\right]+@staff+.\left[A-Z\right]\{2,4\} \\ \rightarrow dLOSS : Internal \\ \left[A-Z0-9\_.-\right]+@inc+.\left[A-Z\right]\{2,4\} \\ \rightarrow dLOSS : External \end{gathered} \tag{3}$$

Applying semantic lifting on the data listed in Table 4, we are able to derive Table 8, where an event is rewritten as `Internal` when the operation is performed by an internal user, otherwise the event is considered `External`.

We can again apply Bayesian inference to quantify the incidence between an internal or external user and P in the Equation 2, while the resulting inference is reported in Table 9.

Table 8. Violation/Non-Violation Prescriptive Data Correlation.

Prescriptive Data Relation	Internal	External	Total
Violation (P_v)	1	2	3
Non Violation ($\overline{P_v}$)	5	0	5
Total	6	2	8

Table 9. Inference.

Share	$\Pi(P_v\|Share)$	$\Pi(\overline{P_v}\|Share)$	Δ
Internal	0.2	0.8	-0.6
External	1	0	1

The result obtained can be measured in terms of *support* and *confidence*, as seen in Equation 4[1], where *support* can be defined as the ratio between the total number of traces in a data set and the traces containing P and P_d, while *confidence* is the ration between the number of traces containing P and P_d and the number of traces containing P_d.

$$Supp = \frac{P \cup P_d}{D}; \quad Conf = \frac{P \cup P_d}{P_d} \tag{4}$$

Confidence can be interpreted as an estimation of the probability $\Pi(P|P_d)$. It then results that Semantic Lifting cannot improve the *confidence* of our incidence analysis while it has positive effects on *support*.

6 Conclusions

Process mining is a process management technique that extracts knowledge from the events and the workflow log recorded by an information system. In this work we showed how standard process mining techniques on events and workflow log can be combined with semantic lifting procedures in order to discover new and more precise process models.

We started defining a case study taken from a standard Data Loss scenario as shown by the log records of Table 2. We built a Dependency/Frequency table based on the successor relation, from which defined a violation constraint for our context and we represented it in Equation 1. Then we contestualized the constraint, building two sets of events, where we respectively collected the Violating and the Non Violating events, as shown in Table 5 and in Figure 8. Finally we computed the inference.

[1] The argument of *Supp*() is a set of preconditions, and thus it becomes more restrictive as it grows, for this reason the union of several preconditions must not be interpreted as a logical disjunction but as a logical conjunction.

This is just a first step to show the feasibility of this approach. As a future work we plan to investigate how to automatize the whole process by exploiting the usage of RDF as a modeling language.

Acknowledgment

This work was partly funded by the Italian Ministry of Economic Development under the Industria 2015 contract - `KITE.it` project.

References

1. der Aalst, W.V.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer Heidelberg, Berlin (2011)
2. Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.: Fundamentals of Business Process Management. Springer Heidelberg (2013)
3. Laguna, M., Marklund, J.: Business Process Modeling, Simulation and Design, Second Edition. CRC Press (2013)
4. Surajit, C., Umeshwar, D., Vivek, N.: An overview of business intelligence technology. Commun. ACM **54**(8) (2011) 88–98
5. Vergidis, K., B., A.T., Majeed: Business process improvement using multi-objective optimisation. BT Technology Journal **24**(2) (2006) 229–235
6. van der Aalst: Business process management: a comprehensive survey. ISRN Software Engineering 2013 (2013)
7. van der Aalst Wil MP: Business process management: A comprehensive survey. ISRN Software Engineering **2013** (2013)
8. Ellis, C.: Information control nets: a mathematical model of office information flow. In: Proceedings of the Conference on Simulation, Measurement and Modeling of Computer Systems, ACM Press (1979) 225–240
9. Röglinger, M., Pöppelbuß, J., Becker, J.: Maturity models in business process management. Business Process Management Journal **18**(2) (2012) 328–346
10. Arigliano, F., Azzini, A., Braghin, C., Caforio, A., Ceravolo, P., Damiani, E., Savarino, V., Vicari, C., Zavatarelli, F.: Knowledge and business intelligence technologies in cross-enterprise environments for italian advanced mechanical industry. In: Proceedings of the 3rd International Symposium on Data-driven Process Discovery and Analysis (SIMPDA 2013), Riva del Garda (TN), CEUR-WS.org (2013) 104–110
11. Colombo, A., Damiani, E., Frati, F., Oltolina, S., Reed, K., Ruffatti, G.: The use of a meta-model to support multi-project process measurement. In: Proceedings of 15th Asia-Pacific software engineering conference (APSEC 2008), Beijing, China (2008) 503–510
12. Clair, C.L., Cullen, A., Keenan, J.: Use a metrics framework to drive bpm excellence. http://www.forrester.com/Use+A+Metrics+Framework +To+Drive+BPM+Excellence/fulltext/-/E-RES82161 (September 2012)
13. Azzini, A., Ceravolo, P.: Consistent process mining over big data triple stores. In: Proceedings of the IEEE International Conference on Big Data, June 27-July 2, 2013, Santa Clara Marriott, CA, USA, IEEE Publisher (2013) to appear
14. Buijs, J.: Mapping data sources to xes in a generic way, master's thesis (2010)

15. Baier, T., Mendling, J.: Bridging abstraction layers in process mining by automated matching of events and activities. In Daniel, F., Wang, J., Weber, B., eds.: Business Process Management. Volume 8094 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2013) 17–32
16. Kehrer, T., Kelter, U., Taentzer, G.: A rule-based approach to the semantic lifting of model differences in the context of model versioning. In: Automated Software Engineering (ASE) 2011 26th IEEE/ACM International Conference on. (2011) 163–172
17. Nicola, A.D., Mascio, T.D., Lezoche, M., Tagliano, F.: Semantic lifting of business process models. 2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops **0** (2008) 120–126
18. Aalst, W.V.D.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer (2011)
19. Papazoglou, M., Heuvel, W.V.D.: Business process development life cycle methodology. In: Communications of the ACM. (2007) 79–85
20. Maggi, F.M., Francescomarino, C.D., Dumas, M., Ghidini, C.: Predictive monitoring of business processes. In: CAiSE. (2014) 457–472
21. van der Aalst, W., Schonenberg, M., Song, M.: Time prediction based on process mining. Information Systems **36**(2) (2011) 450 – 475 Special Issue: Semantic Integration of Data, Multimedia, and Services.
22. Ruta, D., Majeed, B.: Business process forecasting in telecom industry. In: GCC Conference and Exhibition (GCC), 2011 IEEE, IEEE (2011) 389–392
23. Kang, B., Kim, D., Kang, S.H.: Real-time business process monitoring method for prediction of abnormal termination using knni-based lof prediction. Expert Syst. Appl. **39**(5) (April 2012) 6061–6068
24. Folino, F., Guarascio, M., Pontieri, L.: Discovering context-aware models for predicting business process performances. In: On the Move to Meaningful Internet Systems: OTM 2012. Springer (2012) 287–304
25. Pika, A., van der Aalst, W.M., Fidge, C.J., ter Hofstede, A.H., Wynn, M.T.: Predicting deadline transgressions using event logs. In: Business Process Management Workshops, Springer (2013) 211–216
26. Suriadi, S., Ouyang, C., van der Aalst, W.M., ter Hofstede, A.H.: Root cause analysis with enriched process logs. In: Business Process Management Workshops, Springer (2013) 174–186
27. Li, M., Liu, L., Yin, L., Zhu, Y.: A process mining based approach to knowledge maintenance. Information Systems Frontiers **13**(3) (2011) 371–380
28. Knowledge and Business Intelligence Technologies in Cross-Enterprise Environments for Italian Advanced Mechanical Industry. (2013)
29. Hayes, P., McBride, B.: Resource description framework (rdf). `http://www.w3.org/` Date: 2004.
30. Carroll, J., Bizer, C., Hayes, P., Stickler, P.: Named graphs. Journal of Web Semantics **3**(3) (2005)
31. Prudhommeaux, E., Seaborne, A.: Sparql query language for rdf. `http://www.w3.org/` Date: 2008.
32. Leida, M., Majeed, B., Colombo, M., Chu, A.: Lightweight rdf data model for business processes analysis. Data-Driven Process Discovery and Analysis Series: Lecture Notes in Business Information Processing **116** (2012)
33. Parmenter, D.: Key performance indicators (KPI): developing, implementing, and using winning KPIs. John Wiley & Sons (2010)
34. Richardson, R., Director, C.S.I.: Csi computer crime and security survey. Computer Security Institute (2008)

35. Van der Aalst, Wil, T.W., Maruster, L.: Workflow mining: Discovering process models from event logs. Knowledge and Data Engineering **IEEE Transactions on 16.9 (2004)** (2004) 1128–1142
36. Van Der Aalst, W., Van Hee, K.: Workflow management: models, methods, and systems. MIT press (2004)
37. Van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow mining: a survey of issues and approaches. Data Knowl. Eng. **47**(2) (November 2003) 237–267
38. Smyth, P., Goodman, R.M.: Rule induction using information theory. Knowledge discovery in databases **1991** (1991)
39. Hert, M., Reif, G., Gall, H.C.: A comparison of rdb-to-rdf mapping languages. In: Proceedings of the 7th International Conference on Semantic Systems. I-Semantics '11, New York, NY, USA, ACM (2011) 25–32