ENABLING RISK-AWARE ENTERPRISE MODELING USING SEMANTIC ANNOTATIONS AND VISUAL RULES

Benedikt Pittl
University of Vienna, Vienna, Austria, benedikt.pittl@univie.ac.at

Hans-Georg Fill
University of Bamberg, Bamberg, Germany, hans-georg.fill@unibamberg.de

Gerald Honegger
University of Vienna, Vienna, Austria, a1127222@unet.univie.ac.at

ENABLING RISK-AWARE ENTERPRISE MODELING USING SEMANTIC ANNOTATIONS AND VISUAL RULES

Research paper

Benedikt Pittl, University of Vienna, Vienna, Austria, benedikt.pittl@univie.ac.at

Hans-Georg Fill, University of Bamberg, Bamberg, Germany, hans-georg.fill@uni-bamberg.de

Gerald Honegger, University of Vienna, Vienna, Austria, a1127222@unet.univie.ac.at

Abstract

The engagement in professional risk management is today a fact for most large organizations. In order to satisfy regulation and auditing requirements, an important step thereby is the identification and documentation of risks in an organization and the definition of measures for their mitigation. Thereby, the use of enterprise models provides the foundation for a systematic and holistic analysis of processes, organizational structures and IT systems. In the approach at hand we build upon the SeMFIS approach for semantic annotations of enterprise models with concepts from an OWL2 ontology. By providing an ontology for representing risks and mitigation measures, this additional information can be represented through annotations in arbitrary types of enterprise models without having to adapt the originally used modeling language. In addition, the approach provides a visual modeling language for representing rules according to the SWRL specification. This permits to process the semantic information provided by the annotations. The usage of the approach is illustrated through an example from the domain of risk-aware business process management. Upon the representation of risks in business processes using the semantic annotation approach, it is shown how SWRL rules can be used to automatically generate configurable risk reports.

Keywords: Risk management, Enterprise modeling, semantic annotation, ontology, rules.

1 Motivation

Risk management is a well-known but still highly relevant topic for enterprises. In the academic literature, risk management is considered as a necessity to comply with standards, audits and regulations as well as an instrument for gaining competitive advantage, c.f. for example (Strecker, Heise, and Frank 2011), (Jakoubi, Tjoa, and Quirchmayr 2007) or (Fill 2012). The importance of risk management is also reflected in industry-related literature. For example, a study by the Boston Consulting Group (Grasshoff et al. 2016) found that the number of regulations and norms in the banking sector is still continuously growing. Widely known agreements like “Basel I, II and III” are only a subset of the regulations to which banks have to comply today. Therefore, banks are forced to implement a risk management which is flexible enough to cope with unexpected revisions and introductions of norms and regulations. Insurance companies face similar problems (Accenture 2016). In a report by (PwC 2016) it is emphasized that risk management is nowadays also regarded as an instrument for gaining competitive advantage. They consider risk management as an important instrument for enterprises to succeed in an economic environment where changes occur faster than ever before. It is shown that 66% of asked CEOs see more threats than opportunities in the current economic environment (PwC 2016). So-called risk agility is a key characteristic of enterprises to survive. In (PwC 2016) “risk agility” is defined as the ability to alter and adapt risk management infrastructure to respond quickly to changing markets, customer preferences or market dynamics. So, it is obvious that risks need to be considered in the processes of an enterprise to achieve the required short response time to changes in the environment. Suriadi et al. (2014) go one step further and envision a runtime migration of risks to business processes. For the management of such business processes the term risk aware business process management is used (Suriadi et al. 2014). The scientific community has introduced a lot of approaches for such a risk aware business process management varying in terms of goals, functionality and scope (Suriadi et al. 2014). For example, (Strecker, Heise, and Frank 2011) introduced a modeling language for IT risks which is connected with the underlying process models. This model-based approach shall enhance IT risk assessment. The ROPE methodology introduced in (Tjoa, Jakoubi, and Quirchmayr 2008) also encompasses a process model connected with a risk model as well as a simulation environment to assess the incidence of a failure. Contrary, the authors of (Sadiq, Governatori, and Namiri 2007) use the Formal Contract Language (FCL) for extending processes with rules in order to comply with risk control guidelines. Further approaches will be described in the related work section.

In (Fill 2012) we presented an approach for simulating the execution of business processes which is influenced by risks. This basic procedure of the approach was the following: we executed the process several times whereby the execution time of an activity was modified depending on its assigned risks before each run. We assigned the risks to the activities of the business process using annotations to avoid modification of existing models. This is because we assume like (Rosemann 2006) that an enterprise has already thousands of enterprise models. Hence, using a new modeling language considering risks would require to remodel or migrate the existing models to the new modeling language. In the paper at hand, we build upon this assumption and extend our previous approach. Hence, we present a novel way for risk-aware enterprise modeling. Instead of using technologies based on proprietary specifications for rules and ontologies for business process simulation as it has been done in (Fill 2012), we are now reverting to semantic annotations based on open international standards for creating risk reports which are considered as a key element of risk management in the literature, e.g. (Segal 2011). Risk reports permit to document risks in enterprises together with measures for their mitigation as well as quantitative information about the risks such as incidence probabilities and impacts. To accomplish this, we introduce a novel visual language for the Semantic Web Rule Language (SWRL) that will be used for configuring the generation of risk reports. The SWRL recommendation was designed as the rule language of the semantic web (Horrocks et al. 2004) and can be used for describing business rules (Meech 2010). Due to its openly accessible specification it is today widely used for specifying rules.

Although the approach we will describe is applicable to arbitrary types of enterprise models, we will here focus on business process models as one of the most widely researched type of enterprise models for dealing with risks, cf. (Suriadi et al. 2014). Moreover, we assume that an enterprise already uses a generic or enterprise specific knowledge base for storing information about risks. The usage of a risk knowledge base for analyzing the risk awareness of business processes had been suggested by (Alhawari et al. 2012) and was realized in e.g. (Fill 2012). It typically contains a classification of risks as well as a description of them.

The contributions of the paper are: (i) a concept for modeltype-independent risk annotations of enterprise models using open semantic web standards, (ii) design of a SWRL modeling language for the visual specification of rules for generating risk reports, (iii) design and implementation of the approach on the SeMFIS platform, and (iv) a first validation of the approach in a use case.

The remainder of the paper follows: Section 2 summarizes the related work and state-of-the-art in the field of risk aware business process management. In section 3 we describe the concept of our approach together with implementation details in section 4. The use case is described in section 5 followed by a discussion section. The paper ends with a conclusion in section 7.

2 Related Work

In this section, we introduce relevant concepts developed by the scientific community for risk aware business processes, concepts for the integration of rules and business processes as well as simulation techniques for risk aware processes.

The scientific community developed several modeling methods for risk management. In (Strecker, Heise, and Frank 2011) a modeling method for IT risks was developed called RiskM, which is based on the Multi-Perspective Enterprise Modeling language (MEMO). RiskM allows to model processes, IT assets, strategies and goals. The core RiskM modeling procedure is two-stepped. First, the IT-assets are modelled. For example, an IT-assets model represents an ERP-System containing an ERP-Server. Second, the modeling method allows to assign concrete risks to IT assets which may have an effect on other IT assets. E.g., an ERP-Server may have the risk of a hard disc error while the ERP-System may have as risk of "System unavailability". A hard disc error in the ERP-Server causes a system unavailability leading to a breakdown of the whole ERP-System. In the description of the approach, no algorithms or mechanisms were introduced. The risks have to be modeled and cannot be imported from an external knowledge base. Another modeling method was introduced in (Fill et al. 2007). The authors designed a modeling method for a so-called Integrated Enterprise Balancing. Precondition of an integrated enterprise balancing is a common database in the domains of risk, return, regulation and reporting. The authors introduced a modeling method to achieve such a common data structure. Thereby, activities of a process are connected to risks which are in turn connected to events. So, the final model shows which activities of a process bear certain risks. The events which are connected to the risks have a certain probability of occurrence as well as an impact factor.

Several research groups investigated the integration of risks into business processes. The authors of (Zur Muehlen and Ho 2005) introduced a risk taxonomy, identified risks in Business Process Management, classified risks and tried to map Business Process Management risks to their risk taxonomy. However, a general approach defining how risks of a domain specific processes can be handled or analyzed was not presented. The authors of (Conforti et al. 2016) describe a risk management approach used during the execution of processes called PRISM. The main idea of their paper is that identical process instances have the same risk exposure. Sensors (physical as well as non-physical) are used to get information about the process instance and to assess the risk exposure. In cases in which a sensor detects information regarding a risk, the information is cascaded across all currently running instances. Weiss and Winkelmann (2011) introduced a risk view for the semantic business process modeling language (SBPML). Their approach was specifically designed for banks and so its application is limited for other domains. The survey paper (Suriadi et al. 2014) describes a detailed comparison of different approaches in the domain of risk aware business process management. Thereby, the authors distinguish between three main categories. The approaches belonging to the first category handle risks in business processes at design time, the second category is for approaches handling risk at runtime while the last category contains all approaches which handle risks after process execution (post-execution). The approach we will introduce in this paper belongs to the design time category. The authors of (Rosemann and Zur Muehlen 2005) introduced a four-stepped procedure fostering the integration of risks in business process models. In the first step risks are identified, then the processes related to the risks are determined. In the third step alternative process configurations are created which are compared in the final step. The approach was refined in (Neiger et al. 2006). The survey presented in (Carnaghan 2006) analyzes different representation methods of processes in the context of risk auditing. Thereby it was analyzed how far widely used conventions help to understand the underlying process risks. In (Xie et al. 2011) a supply chain management approach was introduced called SCRMP which supports supply chain risk identification, measurement, assessment, evaluation and migration. SCRMP is a pure procedure model without technical integration issues.

Risk identification is important but usually not sufficient for enterprises to capture the consequences of risk incidences. To overcome this shortcoming, simulation approaches for risk management have been conceived. For example, (Jakoubi, Tjoa, and Quirchmayr 2007) introduced the ROPE (Risk-Oriented

Process Evaluation) methodology. This methodology is based on three basic models: a business process model, a CARE model and a TIP model. An activity of the business process model has a reference to elements of the CARE (Condition, Action, Resource and Environment) model describing the required resources. Threats affect one or more CARE elements. In the TIP (Threat Impact Process) model, the effects of threats are described as well as the counter measurements. The ROPE methodology also encompasses a path simulation as well as a simulation for assessing the impact of incidents so that the efficiency of countermeasures can be tested. In (Fill 2012) the author used semantic annotations for assigning risks from a frame ontology to a business process. The models including the annotations were processed via Java with JESS rules. The rules generate ADOxx code (ADOscript) for running an ADOxx capacity simulation to assess the impact of risks.

The integration of business process and rules forms the last section of our related work analysis. The trade-off as well as the connection between business rules and business process models is still a controversial topic. Both describe policies and guidelines of an enterprise (Zur Muehlen and Indulska 2010). Zur Muehlen and Indulska (2010) did a representational analysis of rules and process modeling languages. Thereby, the authors showed that the combination of a rule language with a process language leads to better results regarding representational completeness. Wang, Indulska, and Sadiq (2014) also investigated how rules and business processes can be integrated. They argue that rules are not visually representable leading to lower understandability. Often, modelers try to create rules with business process model constructs leading to complex processes which do not reflect exactly the intended behavior. Betz, Hickl, and Oberweis (2011) developed a new risk aware process modeling language including a simulation mechanism. However, an integration of already existing models is missing. Similarly, the business process simulation environment introduced in (Betz, Hickl, and Oberweis 2011) offers generic simulation mechanisms. However, integration with already existing models is not foreseen. In the FIT project, a modeling method for a visual SWRL rule language was created (Leutgeb et al. 2007). The intention of the authors was to model business rules and map them to the process model elements. The export of the rules model to a machine-readable format was however not discussed. The authors of (Bork and Fill 2014) discussed different formalisms for models.

In summary, it can be concluded that the integration of risks on business process level has been investigated by several research groups. However, most of the existing work either introduces a pure modeling approach or a pure management approach describing how risk management should be considered during the business process management lifecycle. Further, the majority of approaches so far did not consider the application to arbitrary types of enterprise models. What is also missing so far is the integration of rule-based approaches and modeling approaches that are adequate for business users. Although some approaches exist for the visual modeling of rules, they have not been made available to the research community, nor have they been applied to risk management. We intend to overcome this by providing a visual modeling language for rules based on open standards that can be integrated with arbitrary types of enterprise models and that will be made publicly available.

3 Risk-Aware Enterprise Modeling

In previous approaches, enterprise models and risk management have often been regarded as two distinct fields (Suriadi et al. 2014). In the approach we describe in the following, we integrate these two fields. The underlying assumption is that both fields refer to essential parts of enterprise knowledge that need to be treated in equal manner. This also corresponds to the view that is taken by industry today where risk management is seen as an opportunity for gaining competitive advantage (PwC 2016). One of the outcomes of this integration will be the provision of according risk reports for enterprise models. The integration is accomplished through the process shown in figure 1.

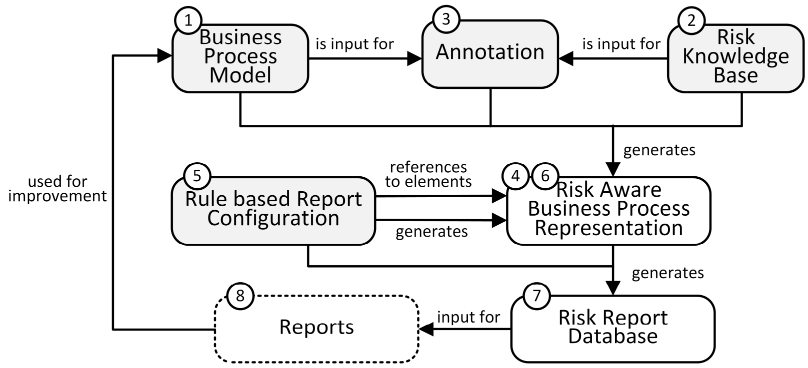


Figure 1. Conceptual relationships of the approach (grey boxes represent models of our modeling language)

The numbers in the following enumeration refer to the steps shown in Figure 1. (1) We assume, that enterprises have already business process models. (2) Further, enterprises need a risk knowledge base containing a description of risks which may be relevant for the elements of the business process model. (3) In our approach, we connect the business process models with the risks in the risk knowledge base using annotations. An annotation describes for example that an activity a_1 from the business process model is exposed to the risk r_1 from the risk knowledge base. Annotations are a separate information chunk referencing to business process elements. Hence, due to the usage of annotations we need not to modify existing business process models nor the underlying modeling language. (4) The business process model, the annotation and the risk knowledge base are merged to a risk aware business process representation. Thereby, the information stored in the annotation is used to map business process model elements with risks in the risk knowledge base. (5) The so created risk aware business process contains already most of the risk relevant information. Additionally, rules can be used to enrich or modify the risk aware business process and to inference further information. In our case, we use rules to enrich and configure the risk aware business process in order to generate reports. (6) We store these rules together with the risk aware business process because rules reference elements in this representation.

So, the resulting risk aware business process representation contains - after a merge process - the information of the business process model, of the risk knowledge base, the annotations of the business process model elements and the rules. (7) Executing the rules on the risk aware business process leads to a so-called risk report database. This artefact contains all the information which is necessary to create risk reports. (8) So, before reports can be created, the risk report database has to be queried. Risk reports itself should help the risk management to improve business process models described in step (1).

We realized our approach using a pure model-based technique. Therefore, we need models for the business processes, the annotations, the risk knowledge base and the rules for report generation. These models are represented as grey boxes in Figure 1. For representing business processes, existing languages such as BPMN, EPC, or ADONIS BPMS can be used. For the annotations, we use semantic annotation models as described in (Fill 2011). For the risk knowledge base, we revert to OWL ontology models as described in more detail in section 4.

In order to ease the design and understanding of rules for supporting business users, we decided to create a visual modeling language for rules according to the most recent and widely used SWRL (Semantic Web Rule Language) standard. Our goal was to create a visual modeling language that can be integrated with other enterprise modeling languages and which is compatible with rule-based applications e.g. the Stanford Protégé platform to enable further processing (Musen 2015; O'Connor et al. 2005).

In SWRL, a rule consists of an antecedent (also called body) and a consequent (also called head). Both, the antecedent and the consequent are based on so-called atoms. These atoms are connected using conjunctions. The SWRL recommendation distinguishes between seven different atoms which can be used in the consequent and the antecedent. According to the W3C recommendation, each atom has parameters (also called terms) which are either variables, individuals or data values. Variables are usually prefixed with a question mark. Individuals and data values are also called constants in the context of SWRL rules. SWRL follows the so called open world assumption. This means that what is not known to be true is simply unknown. SWRL is an extension to OWL (Ontology Web Language), which means that SWRL basically refers to constructs like classes and properties described in OWL. For more information about SWRL we refer the reader to (Horrocks et al. 2004).

3.1 Extension of the SeMFIS Meta Model

For realizing a visual modeling language for SWRL rules, we first investigated the prominent generic rule definition meta model which was published in (Brockmans et al. 2006). In a first iteration, we aimed to merge this meta model with the SeMFIS meta model for semantic annotations and ontologies (Fill 2017). SeMFIS is a modeling method developed with the platform ADOxx (Fill and Karagiannis 2013). Unfortunately, the expressiveness of the generic rule meta model was too low for our purposes including the import and export of SWRL models to standard formats. The reasons were mainly as follows: (i) In the generic rule meta model it is possible to use arbitrary terms such as individuals or variables as parameters for any atom. However, each atom has a well-defined list of allowed parameters, which does not apply to SWRL. (ii) In the generic meta model, each atom has a reference to the class termList. The purpose of a termList seems to trace the position of the parameter in the atom. This does not necessarily make sense for each atom. For example, a class atom in SWRL has only a single parameter. Further, for example a sameAsAtom does not process the position of its parameters. The introduction of this termList as a separate modeling class would lead to a higher (and often unnecessary) modeling effort for the modeler. (iii) It is not described how e.g. anonymous classes, as needed for SWRL, can be handled with this model. For more information about anonymous classes see (McGuinness, Van Harmelen, and others 2004). (iv) The meta model further does not distinguish between all seven different types of atoms which may make import and export to and from models more difficult.

A meta model particularly for SWRL was published in (Leutgeb 2007). Contrary to the generic rule meta model, this meta model distinguishes between all seven types of atoms. For our purposes the meta model had however the following drawbacks: (i) An explicit support for the different types of built-ins which are available in SWRL is misssing. (ii) The publication does not state how anonymous class expressions are handled.

As we did not find an adequate meta model we created our own meta model extension. Our final (simplified) SWRL meta model is shown in the right part of Figure 2. The dashed lines represent interrefs (hyperlinks) which are used to reference other model elements. In more detail, the instances of the classes which have an interref relation to another class in the meta model can reference instances of these classes. Like in the meta model described in (Brockmans et al. 2006), we have modeled the terms as separate classes which we consider to be more convenient for modelers than a solution based on attributes. Further, we can declaratively avoid modeling errors like using data variables as parameters for atoms which do not have data variables as parameters. Like in the meta model described in (Leutgeb 2007) we have modeled each atom as separate class. However, we split the data range atom into two separate atoms. The authors of the SWRL specification ("XML Syntax for SWRL" 2016) state that the datarange element either represents a list of data values or a datatype. Datarange atoms which contain a datatype or data values are semantically related, but their syntax is completely different. Hence, we decided to split the datarange atom in to a datarange atom class containing a datatype and a datarange atomlist class containing a list of datavalues.

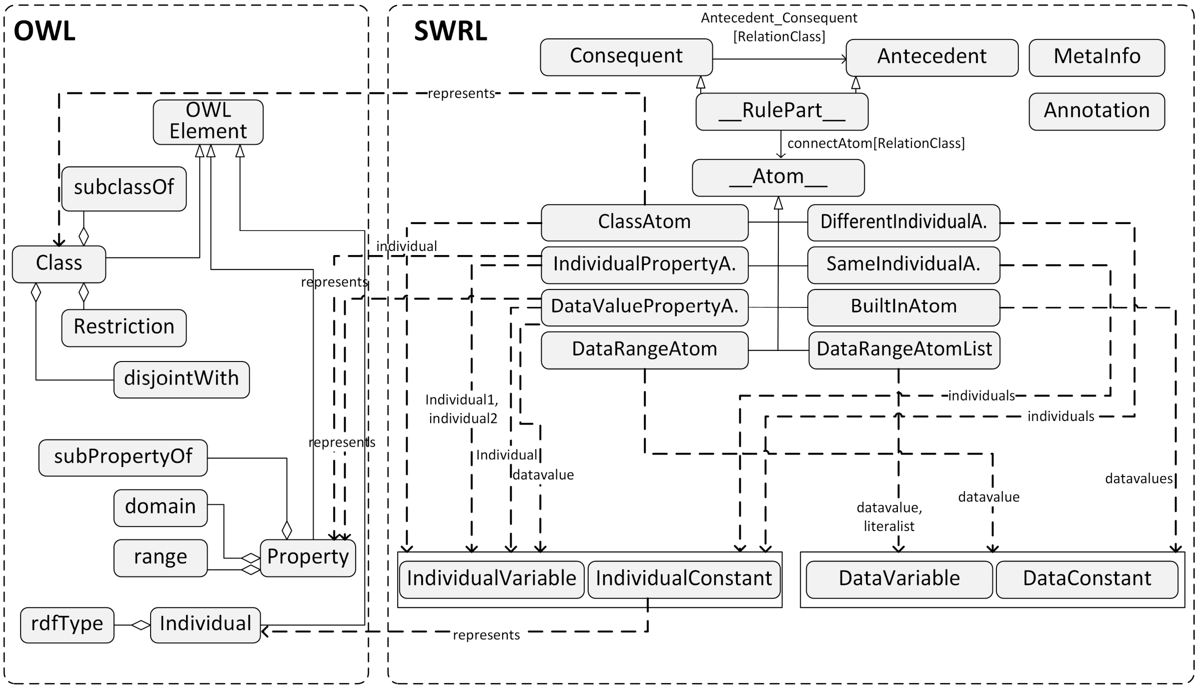


Figure 2. Excerpt of the meta model for SWRL integrated with the OWL metamodel of SeMFIS

With the meta model shown in Figure 2 we describe two visual modeling languages: one for SWRL models and one for OWL models. A challenge was the integration of the SWRL meta model with the SeMFIS meta model that we decided to use for semantic annotations and ontologies (Fill 2017). In figure 2 we omitted the interrefs in the OWL meta model to keep it readable. The OWL meta model is for example relevant for the IndividualConstant class defined in our SWRL meta model. An instance of type IndividualConstant used in the SWRL model represents an instance of type Individual in the OWL meta model. We identified an alternative for the handling of the IndividualConstant class: Instead of using a class IndividualConstant we could directly use an interref relation from the atom classes to the Individual class in the OWL meta model. However, we decided to keep the class IndividualConstant in the meta model so that resulting SWRL models from our modeling language remain self-contained. If we would delete the IndividualConstant class from the meta model, two instances of an atom class using the same individual would refer to the same instance of type Individual in the OWL model. If the OWL model does not exist, both instances of an atom class would have an empty interref relation. The information that the two instances of an atom class refer to the same individual of type Individual would be lost. In total, there are four classes containing interrefs to the OWL meta model: ClassAtom, IndividualPropertyAtom, DatavaluePropertyAtom and the IndividualConstant. All four classes referencing to the OWL model have a string input field called “name”. This name is used in cases the model is exported but interrefs are missing so that valid SWRL serializations can be created. Our SWRL meta model allows to reuse rule heads and bodies making the modeling even more efficient.

3.2 Visual Notation for SWRL Rule Models

The notation elements of our SWRL modeling language are described in Table 1. We also introduced a Meta Info (containing inter alia a rule name) element as well as an annotation element which we did not include in the table to save space. Moreover, we created two relation classes. The first relation connects the antecedent and the consequent while the second relation connects the antecedent as well as the consequent with the atoms.

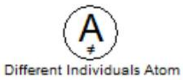
	The antecedent (condition) of a rule		The consequent of a rule
	Class atom e.g. Person(?x)		Individual-valued Property Atom e.g. hasRisk(?x,?y)
	Datavalued Property Atom e.g. hasName(?x,?y)		Datarange Atom with datatype
	Different Individuals Atom e.g. differentFrom(?x,?y)		Same Individuals Atom e.g. sameAs(?x,?y)
	Datarange Atom with literal list		Built-In Atom e.g. greaterThan
	Variable for storing data values		Constant representing a data value
	Variable for storing individual values		Constant representing an individual value

Table 1. Visual Notation for SWRL Rule Models

Figure 3 contains an excerpt of an OWL model, an excerpt of a SWRL model and a textual SWRL rule to illustrate their usage. The left part of the textual rule (before the -->) represents the body, the right part of the rule represents the head. All atoms are connected using a wedge symbol representing a conjunction. The shown rule creates ADOscript code for all “AnnotatedElement” elements which are exposed to a “TechnicalFailure” which has a probability of occurrence higher than 0.9.

The arrows in the figure illustrate the mapping of the rule in textual form to the rule expressed with our visual modeling language. Each atom in the textual rule represents an atom in our visual rule. For example, the atom “AnnotatedElement(?elementVar)” is represented by the class atom element in our visual modeling method. As explained previously, SWRL is an extension for OWL. The OWL model shown in the upper part of Figure 3 was created with SeMFIS. Our class atom “AnnotatedElement” references the OWL class “AnnoatedElement” using an interref which is represented with a dotted line. The relation between class atoms in SWRL and classes in OWL was already shown in Figure 2. For a better readability of the figure we have not shown the other interrefs.

If a SWRL model or ontology model is imported, a specifically-designed layouting algorithm is applied on it. This prevents the import algorithms from placing all model elements in one place in the model, which would be the standard procedure. Instead, the different model elements will be placed in a way so that the model is clear and comprehensible for the user.

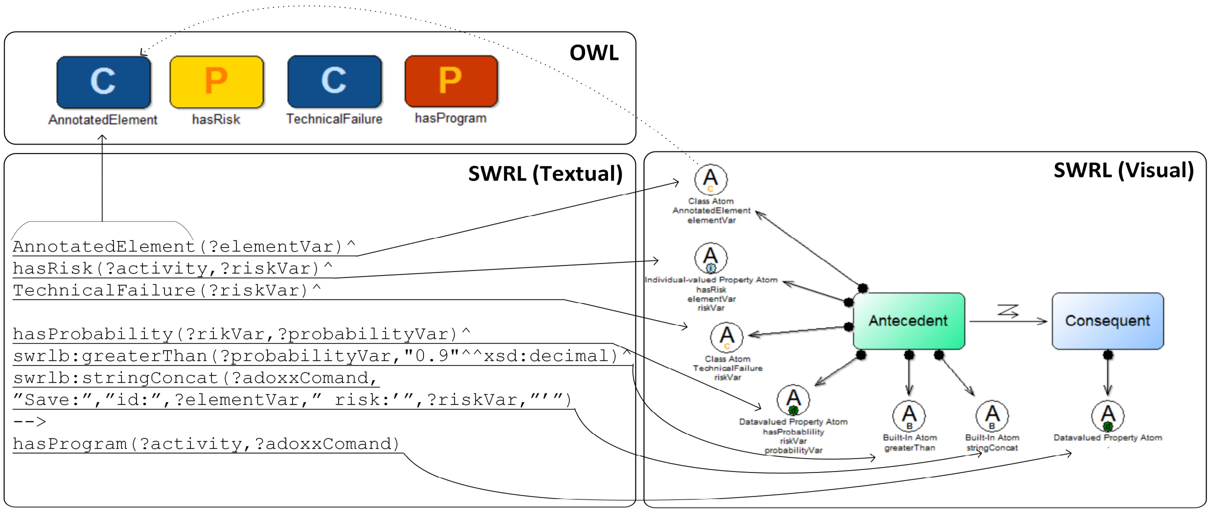


Figure 3. Mapping between SWRL rule in text form and its visual representation (simplified)

4 Technical Implementation Using SeMFIS

As had already been mentioned above, we revert to ontologies in W3C OWL format for describing the risk knowledge bases. As the content of the risk knowledge base is usually a description of possible risks including their impact, probabilities and relations to each other, an ontology seems to be an adequate choice for its representation. In our case, the web ontology language OWL has been chosen for the following reasons. First, it is already part of the SeMFIS modeling method that we reverted to for the semantic annotations. Second, as SWRL extends OWL we have no integration problems between the rules which we create with our SWRL modeling language and the knowledge base.

In this section, we want to describe the technical details of our approach. Due to the space limitations, we can only summarize the most important aspects. Figure 4 is an extension of Figure 1 which emphasizes the implementation details. Again, the following numbers refer to the numbers shown in Figure 4. First, business process, annotation, and ontology models (1,2,3,4) are created using the SeMFIS platform.

After the models are obtained, we translate them in two ontologies. (5) Our application can export the business process model, the risk ontology model as well as the annotation model to an RDF ontology. For this we created as a byproduct a novel RDFizer for conceptual models. Buchmann and Karagiannis (2016) had already introduced an RDFizer for translating conceptual models into RDF format. Their RDFizer is highly customizable but requires special attributes in the model elements which were resolved to RDF triples. Contrary, our RDFizer is a pure model based RDFizer for RDFizing models which does not require to modify the existing modeling method. The model based annotation mapping is sufficient to create RDF triples. Our model based RDFizer is not limited to the risk management approach presented in the paper at hand. Indeed, it can be used for any so-called model aware information system to leverage the information out of conceptual models such as intended by (Buchmann and Karagiannis 2016). (6) The rules as well as the risk ontology are exported to an OWL2 ontology. Thereby, we support two formats: OWL2-XML as well as SWRL-XML. The ontologies are independent from each other to keep our implementation as generic as possible. Model aware information systems may process only the RDFized models while classical rule base application may process only the rule based data. For the export (and also for the import) we created XSLT files transforming the internal file format of the models in ADOxx (adoxml) to a standardized file format like OWL2-XML (for the SeMFIS OWL2 Export) or RDF-XML (for the SeMFIS RDFizer). For the handling of anonymous classes, we developed different algorithms for an eligible transformation between ontology models and ontologies in standard file formats.

(7) We merge the two ontologies to a single OWL2 ontology containing both, the rules as well as the model based data. Therefore, we used the Java based OWLAPI which is also part of applications like Protégé. (8) Afterwards, we inference the rules using the OWLAPI as well as the SWRLAPI which is contained in the Protégé project. The rules are used for setting up the ontology so that risk reports can be created with SeMFIS. (9) The resulting ontology is called "Risk Report OWL Ontology Database". It is still an OWL2 ontology and so we query it with SQWRL (Semantic Query-Enhanced Web Rule Language) (O'Connor and Das 2009) to get the relevant data for creating ADOxx commands to trigger the risk report generation as will be shown in our use case.

The dotted boxes in Figure 4 are components of (external) rule based systems. In our case shown in the following example, we created a Java base rule base application interacting with SeMFIS. The dashed boxes represent extension of SeMFIS which we implemented.

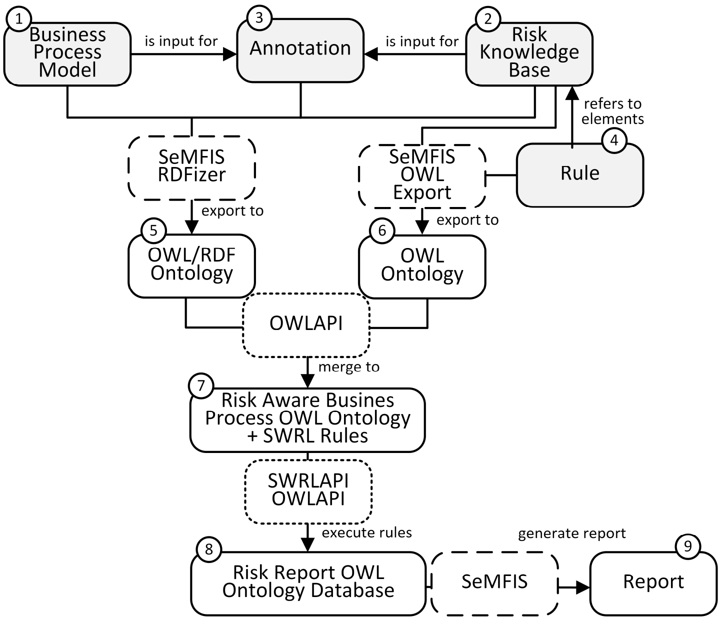


Figure 4. Technical implementation of the approach

5 Use Case for Risk-Aware Business Process Management

In this section, we describe our approach using a minimal but illustrative example to clarify our approach. First, we describe the underlying risk ontology in more detail which is the basis for the introduced use case.

5.1 Risk Management Ontology

In (Fill, 2012) a knowledge base for risks using a frames ontology was created. The used risk classification is based on the "IT-Grundschutz catalogue" published by the German Federal Office for Information Security ("BSI - IT-Grundschutz-Katalog" 2016). The generic risk classification consisting of "Elementary Threats", "Force Majeure", "Organizational Shortcomings", "Human Failure", "Technical Failure" and "Deliberate Acts" results from this catalogue. Each risk is described by its occurrence probability as well as a risk distribution as explained by (Jallow et al. 2007). We reused the frames ontology of (Fill 2012) but we converted it to an OWL ontology resulting into minor differences compared to the frames ontology.

Figure 5 shows the final ontology which we implemented in OWL. The configuration classes are used for gluing the elements of the enterprise models to the risks. The risks listed in the catalogue are subclasses of the class "Risk". Each risk has a risk distribution containing a probability as well as distribution specific properties. For example, the triangular distribution has a lower impact value, a medium impact value as well as a high impact value. For more information about the risk ontology we refer to our previous publication (Fill 2012).

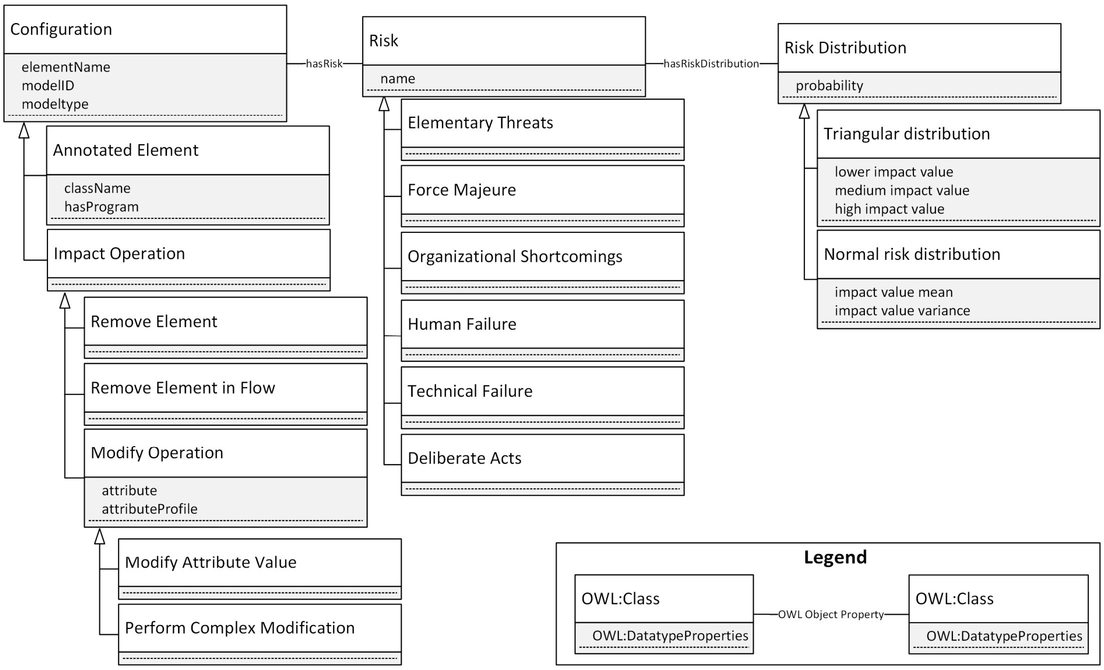


Figure 5. Excerpt of the used OWL2 ontology

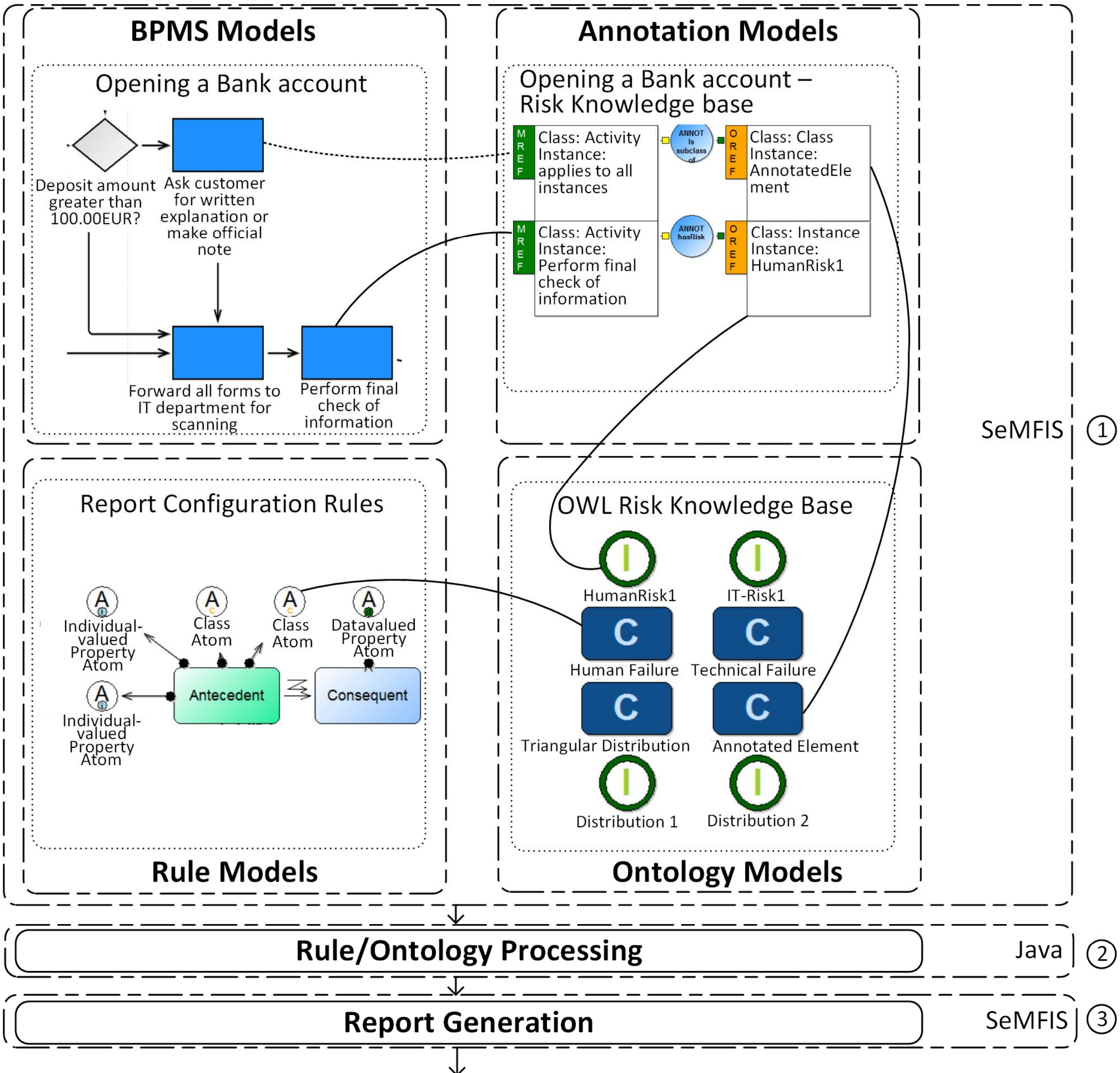
5.2 Sample Process

For the example, we created a business process model in ADONIS BPMS notation describing the opening of a bank as described in ("SeMFIS" 2017). For more information about the OMiLAB please see (Götzinger, Miron, and Staffel 2016). The process model is shown in the upper left corner in Figure 6. The shown excerpt contains three activities and one decision. The risk ontology shown in Figure 5 was realized as an OWL model. An excerpt is shown in the right lower corner in Figure 6. It contains two risks: the "Human Risk 1" and the "IT Risk 1". Both have a triangular distribution. Our goal in this use case is to annotate the business process with the risks and to generate a risk report.

The activities of the business process are connected with the elements (risks) o("SeMFIS" 2017)f the risk ontology model using annotations as shown in the right upper corner in Figure 6. Our example shows that e.g. the process activity "Perform final check of information" is annotated with the risk "Human Risk 1". The lines between the model elements (see for example the line between "Perform final check of information" and "Modeling Reference (MREF)") are the previous explained interrefs.

For a flexible annotation, we use different interref types for linking risks to business process activities. So, the dashed links between "Ask customer for written explanation or make official note" and the "Model Reference (MREF)" element means that all instances of the same type (in this case business process activity) share the same annotation. So also e.g. the activity "Perform final check of information" is an "Annotated Element". A solid line like between "Perform final check of information" and "Model Reference (MREF)" means that only the business process activity element has this risk. This example also illustrates the generic appliance of our approach: each enterprise model can be annotated with risks.

For connecting the rule model with the ontology model, we used interrefs. In Figure 6 this is exemplarily shown with the line connecting the class atom with the Technical Failure class of the OWL model. The models are exported to ontologies so that they can be processed with rules created with our visual language. Therefore, we used our Java application (see step 2 in Figure 6). After processing is finished, the resulting ADOxx script is executed for the creation of the risk report (see step 3 in ***Figure 6***).



Activity	Risk	Distribution	Prob. of Occ.	Lower Limit	Upper Limit	Mode
Perform final..	HumanRisk1	Triangular Distribution	0.03	0	1	1
...	...	...	...	...	...	...

Figure 6. Excerpt of the models used for creating the use case (models are screenshots of our modeling method, for better readability we replaced the text)

The simplified structure of a rule for generating risk reports looks like the following:

```
AnnotatedElement(?activity)^ HumanFailure(?risk)^ hasRisk(?activity, ?risk)^
hasRiskDistribution(?risk, ?distribution)^ TriangularDistribution(?distribution)^
hasHighImpactValue(?distribution, ?hasHighImpact) ^ hasMediumImpact-
Value(?distribution, ?hasMediumImpact)^ hasLowImpactValue(?distribution, ?hasLowIm-
pact)^ hasName(?risk, ?riskName)^ hasElementname(?activity, ?elementName)^
swrlb:stringConcat(?adoxxComand, "SAVE ", " id:'", ?elementName, "'", " risk:'",
?riskName, "'") -> hasProgram(?activity, ?adoxxComand)
```

As can be seen, the rule does not create the report directly. Instead, it creates ADOscript code which creates the risk reports. The rule was created using the visual SWRL modeling language which we introduced within this paper.

Figure** 7* shows a screenshot of so a generated risk report document. It contains the annotated business process activity. The activity “Note authorization in Bank-ERP system” was not shown in the excerpt illustrated in ***Figure 6. The generated document can be easily extended and modified.

	A	B	C	D	E	F	G
1	**Risk Report**						
2	Process: Opening a Bank Account						
3							
4	**Business Process Activity**	**Risk Name**	**Risk Distribution**	**Probability of Occurance**	**Lower Limit**	**Upper Limit**	**Mode**
5	Perform final check of information	HumanRisk1	Triangular Distribution	0.03	0	1	1

Figure 7. Screenshot of a risk report generated with our annotation based approach

6 Discussion

In our use case, we created risk reports by using an annotation based approach together with rule models introduced in this paper. However, due to the usage of rules and ontologies which are both based on open standards our approach is not limited to the risk domain. Indeed, the rules can contain any operation that can be expressed in SWRL and the ontologies can be used across several domains. The approach could thus be used for all applications that require the processing of semantic annotations of enterprise models without having to modify an existing modeling language. This could apply to all fields that make heavy use of conceptual models today and where flexibility and agility are highly required to quickly adapt to changing environments, e.g. in enterprise architecture management, business process management or software engineering. Through providing a visual modeling language for the open SWRL standard as well as according import/export functionalities, such rules can be specified without detailed technical know-how of the underlying rule language. One of the next steps in our research will be to evaluate the adequacy of this visual rule modeling language for business users in real-life settings. Additionally, the processing of ontologies has benefits regarding information integration. The information stored in for example existing databases can be converted to RDF using RDFizers like D2RQ and merged with the ontologies generated from the models in order to check compliance or to enrich the already stored information. However, the approach has also some drawbacks. We transform the information stored in models to ontologies to avoid modifications of existing models. The ontology contains all the required information, i.e. which business process elements are exposed to which risk. However, modeling tools like ADOxx cannot handle directly ontologies and so the ontologies need to be processed by other applications (Fill & Karagiannis, 2013). Thus, in our use case example we created a Java based application using the OWLAPI and SWRLAPI which generates ADOscript code which can be processed by ADOxx platform.

7 Conclusion and Outlook

In this paper, we presented an annotation based approach for creating risk aware business process. Due to the usage of semantic web technologies (SWRL, OWL2 and RDF) our approach can be extended easily. For a quick creation and fluent modification of rules we developed a visual modeling language for SWRL rules which we integrated into SeMFIS. To make the rules models executable we implemented several export and import functions to and from standardized file formats. As a by-product, we developed a model-based RDFizer for conceptual models which is (i) more flexible than existing RDFizers for conceptual models and (ii) requires no programming skills or knowledge in the semantic web domain. With the introduced use-case we evaluated our concept by using SeMFIS/ADOxx as a meta modeling platform. In this way it could be shown, that the generation of risk reports can be successfully accomplished by our approach. In our further research, we will extend our approach to using risk annotations during the execution of processes and we will further evaluate the approach through experiments with business users in real-life settings.

References

Accenture. 2016. "Accenture 2015 Global Risk Management Study: Insurance Report Paths to Prosperity." 2016. https://www.accenture.com/_acnmedia/PDF-10/Accenture-Paths-To-Prosperity.pdf (visited on 07/11/2016)

Alhawari, Samer, Louay Karadsheh, Amine Nehari Talet, and Ebrahim Mansour. 2012. "Knowledge-Based Risk Management Framework for Information Technology Project." *International Journal of Information Management* 32 (1): 50–65.

Betz, Stefanie, Susan Hickl, and Andreas Oberweis. 2011. "Risk-Aware Business Process Modeling and Simulation Using Xml Nets." In *2011 IEEE 13th Conference on Commerce and Enterprise Computing*, 349–356. IEEE.

Bork, Domenik, and Hans-Georg Fill. 2014. "Formal Aspects of Enterprise Modeling Methods: A Comparison Framework." In *2014 47th Hawaii International Conference on System Sciences*, 3400–3409. IEEE.

Brockmans, Saartje, Peter Haase, Pascal Hitzler, and Rudi Studer. 2006. "A Metamodel and UML Profile for Rule-Extended OWL DL Ontologies." In *European Semantic Web Conference*, 303–316. Springer.

"BSI - IT-Grundschutz-Katalog." 2016. https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/itgrundschutzkataloge_node.html (visited on 07/11/2016)

Buchmann, Robert A., and Dimitris Karagiannis. 2016. "Enriching Linked Data with Semantics from Domain-Specific Diagrammatic Models." *Business & Information Systems Engineering* 58 (5): 341–353.

Carnaghan, Carla. 2006. "Business Process Modeling Approaches in the Context of Process Level Audit Risk Assessment: An Analysis and Comparison." *International Journal of Accounting Information Systems* 7 (2): 170–204.

Conforti, Raffaele, Sven Fink, Jonas Manderscheid, and Maximilian Röglinger. 2016. "PRISM–A Predictive Risk Monitoring Approach for Business Processes." In *International Conference on Business Process Management*, 383–400. Springer.

Fill, Hans-Georg. 2011. "On the Conceptualization of a Modeling Language for Semantic Model Annotations." In *International Conference on Advanced Information Systems Engineering Workshops, CAiSE 2011, ed. Camille Salinesi and Óscar Pastor,* LNBIP 83 (2): 134–148. Springer.

Fill, Hans-Georg. 2012. "An Approach for Analyzing the Effects of Risks on Business Processes Using Semantic Annotations." In *20th European Conference on Information Systems, ECIS 2012, Barcelona, Spain, June 10-13, 2012*, 111.

Fill, Hans-Georg. 2017. "SeMFIS: A Flexible Engineering Platform for Semantic Annotations of Conceptual Models." *Semantic Web (SWJ)* 8 (5): 747–763.

Fill, Hans-Georg, Anke Gericke, Dimitris Karagiannis, and Robert Winter. 2007. "Modellierung Für Integrated Enterprise Balancing." *Wirtschaftsinformatik* 49 (6): 419–429.

Fill, Hans-Georg, and Dimitris Karagiannis. 2013. "On the Conceptualisation of Modelling Methods Using the ADOxx Meta Modelling Platform." *Enterprise Modelling and Information Systems Architectures-An International Journal* 8 (1).

Götzinger, David, Elena-Teodora Miron, and Franz Staffel. 2016. "OMiLAB: An Open Collaborative Environment for Modeling Method Engineering." In *Domain-Specific Conceptual Modeling*, ed. Dimitris Karagiannis, Heinrich C. Mayr and John Mylopoulos, 55–76. Springer.

Grasshoff, Gerald, Pfuhler Thomas, Grossmann Martin, and Wiemann Markus. 2016. "Risk Report 2015/2016 : White Paper Edition It's Not a Wave: Setting Sail to Master Regulatory Change."

Horrocks, Ian, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, and others. 2004. "SWRL: A Semantic Web Rule Language Combining OWL and RuleML." *W3C Member Submission* 21: 79.

Jakoubi, Stefan, Simon Tjoa, and Gerald Quirchmayr. 2007. "Rope: A Methodology for Enabling the Risk-Aware Modelling and Simulation of Business Processes." In *ECIS*, 1596–1607.

Jallow, A. K., B. Majeed, K. Vergidis, A. Tiwari, and R. Roy. 2007. "Operational Risk Analysis in Business Processes." *BT Technology Journal* 25 (1): 168–177.

Leutgeb, Andrea. 2007. *The Business Rules Method : A Modeling Method for Adaptive Processes - Master Thesis*. Wien: University of Vienna.

Leutgeb, Andrea, Wilfrid Utz, Robert Woitsch, and Hans-Georg Fill. 2007. "Adaptive Processes in E-Government - A Field Report about Semantic-Based Approaches from the EU-Project 'FIT.'" In *ICEIS 2007,* 264–269.

McGuinness, Deborah L., Frank Van Harmelen, and others. 2004. "OWL Web Ontology Language Overview." *W3C Recommendation* 10 (10): 2004.

Meech, Alan. 2010. "Business Rules Using OWL and SWRL." *Advanced in Semantic Computing* 2: 23–31.

Musen, Mark A. 2015. "The Protégé Project: A Look Back and a Look Forward." *AI Matters* 1 (4): 4–12.

Neiger, Dina, Leonid Churilov, Michael zur Muehlen, and Michael Rosemann. 2006. "Integrating Risks in Business Process Models with Value Focused Process Engineering." In *ECIS*, 1606–1615.

O'Connor, Martin, and Amar Das. 2009. "SQWRL: A Query Language for OWL." In *Proceedings of the 6th International Conference on OWL: Experiences and Directions-Volume 529*, 208–215.

O'Connor, Martin, Holger Knublauch, Samson Tu, Benjamin Grosof, Mike Dean, William Grosso, and Mark Musen. 2005. "Supporting Rule System Interoperability on the Semantic Web with SWRL." In *International Semantic Web Conference*, 974–986. Springer.

PwC. 2016. "Risk in Review Going the Distance." https://www.pwc.se/sv/pdf-reports/risk-in-review-going-the-distance.pdf (visited on 07/11/2016)

Rosemann, Michael. 2006. "Potential Pitfalls of Process Modeling: Part A." *Business Process Management Journal* 12 (2): 249–254.

Rosemann, Michael, and Michael Zur Muehlen. 2005. "Integrating Risks in Business Process Models." *ACIS 2005 Proceedings*, 50.

Sadiq, Shazia, Guido Governatori, and Kioumars Namiri. 2007. "Modeling Control Objectives for Business Process Compliance." In *International Conference on Business Process Management*, 149–164. Springer.

Segal, Sim. 2011. *Corporate Value of Enterprise Risk Management: The Next Step in Business Management*. John Wiley & Sons.

"SeMFIS." 2017. http://semfis-platform.org/ (visited on 07/11/2016)

Strecker, Stefan, David Heise, and Ulrich Frank. 2011. "RiskM: A Multi-Perspective Modeling Method for IT Risk Assessment." *Information Systems Frontiers* 13 (4): 595–611.

Suriadi, Suriadi, Burkhard Weiss, Axel Winkelmann, Arthur HM ter Hofstede, Michael Adams, Raffaele Conforti, Colin Fidge, et al. 2014. "Current Research in Risk-Aware Business Process Management: Overview, Comparison, and Gap Analysis." *Communications of the Association for Information Systems* 34 (1): 933–984.

Tjoa, Simon, Stefan Jakoubi, and Gerald Quirchmayr. 2008. "Enhancing Business Impact Analysis and Risk Assessment Applying a Risk-Aware Business Process Modeling and Simulation Methodology." In *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, 179–186. IEEE.

Wang, Wei, Marta Indulska, and Shazia Sadiq. 2014. "Integrated Modelling of Business Process Models and Business Rules: A Research Agenda.". ACIS.

Weiss, Burkhard, and Axel Winkelmann. 2011. "Developing a Process-Oriented Notation for Modeling Operational Risks-a Conceptual Metamodel Approach to Operational Risk Management in Knowledge Intensive Business Processes within the Financial Industry." In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, 1–10. IEEE.

Xie, Charlene, Chimay J. Anumba, Tzong-Ru Lee, Rao Tummala, and Tobias Schoenherr. 2011. "Assessing and Managing Risks Using the Supply Chain Risk Management Process (SCRMP)." *Supply Chain Management: An International Journal* 16 (6): 474–483.

"XML Syntax for SWRL." 2016. http://www.daml.org/swrl/proposal/xmlsyntax.html (visited on 07/11/2016)

Zur Muehlen, Michael, and Danny Ting-Yi Ho. 2005. "Risk Management in the BPM Lifecycle." In *International Conference on Business Process Management*, 454–466. Springer.

Zur Muehlen, Michael, and Marta Indulska. 2010. "Modeling Languages for Business Processes and Business Rules: A Representational Analysis." *Information Systems* 35 (4): 379–390.