

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# UnBBayes: Modeling Uncertainty for Plausible Reasoning in the Semantic Web

Rommel Carvalho<sup>1</sup>, Kathryn Laskey<sup>1</sup>, Paulo Costa<sup>1</sup>, Marcelo Ladeira<sup>2</sup>,  
Laécio Santos<sup>2</sup>, and Shou Matsumoto<sup>2</sup>

<sup>1</sup>*George Mason University  
USA*

<sup>2</sup>*University of Brasilia  
Brazil*

## 1. Introduction

The same assumptions that were essential in the document web are still applied for the Semantic Web (SW). They are radical notions of information sharing, which include [Allemang & Hendler, 2008]: (i) the Anyone can say Anything about Any topic (AAA) slogan; (ii) the open world assumption, i.e. there might exist more information out there that we are not aware of, and (iii) nonunique naming, meaning that different people can assign different names to the same concept. However, the Semantic Web differs from its predecessors in the sense that it intends to provide an environment not only for allowing information sharing but also for making it possible to have the effect of knowledge synergy. Nevertheless, this can lead to a chaotic scenario with disagreements and conflicts.

We call an environment characterized by the above assumptions a Radical Information Sharing (RIS) environment. The challenge facing SW architects is therefore to avoid the natural chaos to which RIS environments are prone, and move to a state characterized by information sharing, cooperation and collaboration. According to [Allemang & Hendler, 2008], one solution to this challenge lies in modeling. Modeling is a simplified abstraction of some real world phenomenon, which, amongst other things, allows the organizing of information for the community use. Modeling supports information sharing in three ways: it provides a means for human communication, it provides a way for explaining conclusions, and it provides the managing of different viewpoints.

There is an immense variety of modeling approaches. In this chapter we will go over a few of these approaches, showing how they can be used and their main limitations related to achieving the full potential of the Semantic Web.

First we will show how to apply Unified Modeling Language (UML) [Rumbaugh et al., 1998] and Entity/Relationship (ER) [Chen, 1976] diagrams for modeling. Then we will present Knowledge Representation and Reasoning (KR&R) [Brachman & Levesque, 2004] and describe how KR&R overcomes some of the limitations of UML and ER. Finally, we present Ontology and the Semantic Web [Berners-Lee, 1999] and discuss how it differs from and moves beyond the previous approaches.

Appreciation is growing within the Semantic Web community of the need to represent and reason with uncertainty. In recognition of this need, the World Wide Web Consortium (W3C) created the Uncertainty Reasoning for the World Wide Web Incubator Group (URW3-XG) in 2007 to identify requirements for reasoning with and representing uncertain information in the World Wide Web. The URW3-XG concluded that standardized representations were needed to express uncertainty in Web-based information [Laskey & Laskey, 2008]. A candidate representation for uncertainty reasoning in the Semantic Web is Probabilistic OWL (PR-OWL) [Costa, 2005], an OWL upper ontology for representing probabilistic ontologies based on Multi-Entity Bayesian Networks (MEBN) [Laskey, 2008].

We will describe PR-OWL and MEBN and show how they contribute to the Semantic Web. Once we have an understanding of the role each modeling approach plays in the Semantic Web we will discuss UnBBayes, an open source, Java-based application based on the PR-OWL/MEBN framework. UnBBayes provides a natural, modeler friendly interface for building probabilistic ontologies and performing reasoning.

This Chapter has three main objectives. First, as described above, we highlight some differences between well-known and used modeling approaches and new ones involving the Semantic Web. Second, we present the main modules of UnBBayes and how to use them to build a probabilistic ontology. Finally, our third objective is to illustrate our approach to model a probabilistic ontology by going through a use case from the Brazilian Office of the Comptroller General (CGU). This probabilistic ontology was developed using UnBBayes. The objective of constructing the ontology is to support fusion of information to detect possible frauds in procurements involving Federal money.

CGU is the Brazilian central body of the internal control system of the federal executive branch. It has, among its responsibilities, the task of inspecting and auditing the Brazilian Government projects and programs with respect to their legality, results, efficacy and efficiency. In Brazil, all contracts with the private sector must be in accordance with the Law N° 8,666/93, also known as the national Procurement Law.

According to [Meirelles, 1996] procurement is the administrative procedure by which the Public Administration selects the most advantageous proposal for a contract in its interest. From the former definition, the conclusion is that the public interest must always be the objective of the procedure. In terms of purchasing with the use of public money, this means that not only must the winner of the procurement process be the best supplier in terms of the price of the good or service supplied, but also in terms of other objectives of the procurement process.

According to [Mueller, 1998], there are many instances where corruption can happen in Public Procurements in Brazil. It goes from a public agency favoring a specific enterprise because the owner is his friend or even worse by receiving some kind of compensation. Collusion is another problem present and hard to prove. Being coordinated from inside, outside, or both, the public procurement process is susceptible to many forms of corruption.

One of the main reasons to being tempted to cheat in public procurements is the fact that the government spends large quantities of money in the market and is a guaranteed payer [Mueller, 1998].

The Brazilian Procurement Law has many articles dedicated to preventing and punishing such instances of corruption. There are two types of penalties stated in Law N° 8,666/93 dealing with this subject. They are administrative actions and penal actions. However, it is not uncommon to see the subject corruption in the press with public servants being accused of such illicit activity.

Since enforcing the law is difficult [Mueller, 1998], public agencies like CGU have been focusing on preventing corruption thus avoiding the need to prove that corruption has occurred. The main focus has been on favoring prevention over punishment.

This is exactly what we are modeling in the use case presented in this Chapter. We try to prevent and detect corruption by analyzing data from the various databases collected by different agencies. It is worth noting that this is still a work in progress. Its use in this Chapter is intended to show the benefits of dealing with uncertainty in the Semantic Web as well as to show how to model this kind of problem. It is not our intention to claim this model as a solution to corruption in Brazilian public procurements.

This Chapter is structured as follows. Section 2 introduces UML and ER. Section 3 presents KR&R and describes how it overcomes some of the limitations of UML and ER. Section 4 describes Ontology and the Semantic Web, and discusses how they differ from the modeling approaches presented in the previous sections. Section 5 summarizes PR-OWL and MEBN as a way to model uncertainty in the Semantic Web. Finally, Section 6 describes UnBBayes main modules by going through a use case being developed at CGU for identifying and preventing fraud in public procurements.

## 2. UML and ER

Before analyzing what we can model in UML and ER, we will define some terms that will be used from now on:

- **Classes** represent concepts, which are understood in a broad sense. For instance, in the procurement domain, the *Goal* class represents a specific objective that needs to be achieved.
- **Instances** are used to represent elements or individuals. For instance, *build a bridge* and *buy 500 chairs* might be specific individuals of the class *Goal*.
- **Relations** represent a link between classes. Although it is possible to represent a relation between many classes at once, the most common is a binary relation, where the first argument is known as the domain of the relation, and the second argument is the range. For instance, classes are usually organized in taxonomies through which inheritance mechanisms can be applied. The binary relation *subclassOf* is used to construct this taxonomy (e.g. *PublicServant* – someone who works for the Government – might be modeled as a subclass of *Person*).
- **Functions** are similar to relations, however, its last element is unique given its preceding elements. For instance, a function *EvaluateEnterprise* applies a set of score rules to compute the final score an enterprise receives when participating in a specific procurement.
- **Formal axioms** are used to model conditions that are always true on the model. It is usually represented by first-order logic sentence that resemble an if statement. For instance, if a person X is member of the committee responsible for procurement Y, then this person X cannot submit a proposal for this procurement Y. These axioms are usually used for [Gómez-Pérez et al., 2005] consistency checking and for inferring new knowledge.

Unified Modeling Language (UML) [Rumbaugh et al., 1998] and Entity/Relationship (ER) [Chen, 1976] diagrams are often used to organize information for community use. Some of the reasons to use these tools are: (i) UML and ER are easy to

understand; (ii) there are standard graphical representations for UML and ER diagrams; and (iii) many CASE tools are available to support development of UML and ER representations.

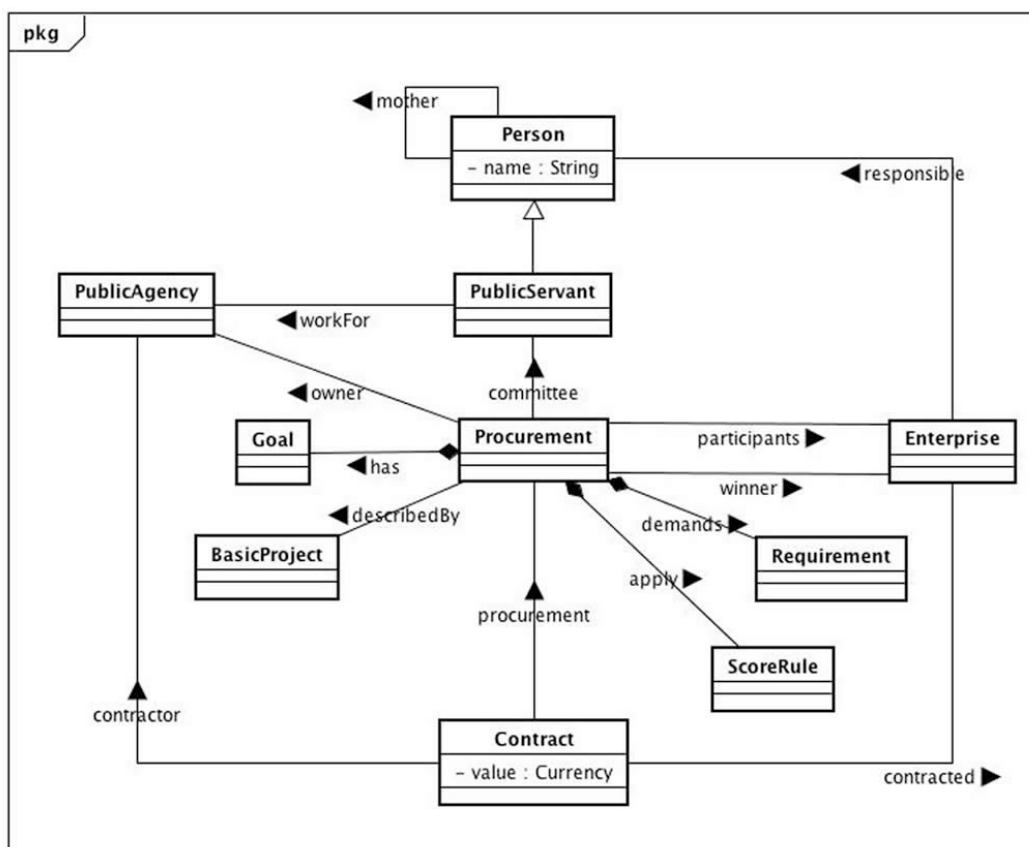


Fig. 1. A class diagram for the procurement domain

UML models can be enriched by adding Object Constraint Language (OCL) [OMG, 1997; Warner & Kleppe, 1998] expressions. OCL is a declarative language for describing rules that apply to UML by providing expressions that do not have the ambiguities of natural language, and avoid the inherent difficulty of using complex mathematics.

In UML, classes are represented with rectangles divided into three parts: the name (top), the attributes (middle), and the operations (bottom). Since operations are not used in the context of the Semantic Web [Gómez-Pérez et al., 2005], we will not deal with them here. The attribute types, possible attribute values, and default values are included in their description. For instance, in Figure 1 *Person* is a class with attribute *name* of type *String*<sup>1</sup>.

Class instances are represented as rectangles divided into two parts. The first part is the name of the instance, followed by “:” and the name of the class it represents. The second part is the attributes of the instance and their respective values. For example, Figure 2 shows four instances, *winner1*, *participant2*, *participant3*, and *participant4*, of the class *Enterprise*.

<sup>1</sup> In UML it is common to use uppercase for classes (e.g., *String*) and lowercase for primitive types and attributes (e.g., *name*).

Concept taxonomies are created through generalization relationships between classes. These are shown on a UML diagram by a solid line extending from the more specific to the more generic class, and ending with a large hollow triangle. In Figure 1, *Person* is a generalization of *PublicServant*, thus it inherits its attribute *name*.

Binary relations are expressed as associations (solid arrows) between classes. In Figure 1 a *PublicServant* works for a *PublicAgency*. However, higher arity relations cannot be represented directly in UML, though we can represent them by creating a class. This class is associated with other classes that represent the relation arguments, as shown in the ternary relation *Contract* in Figure 1.

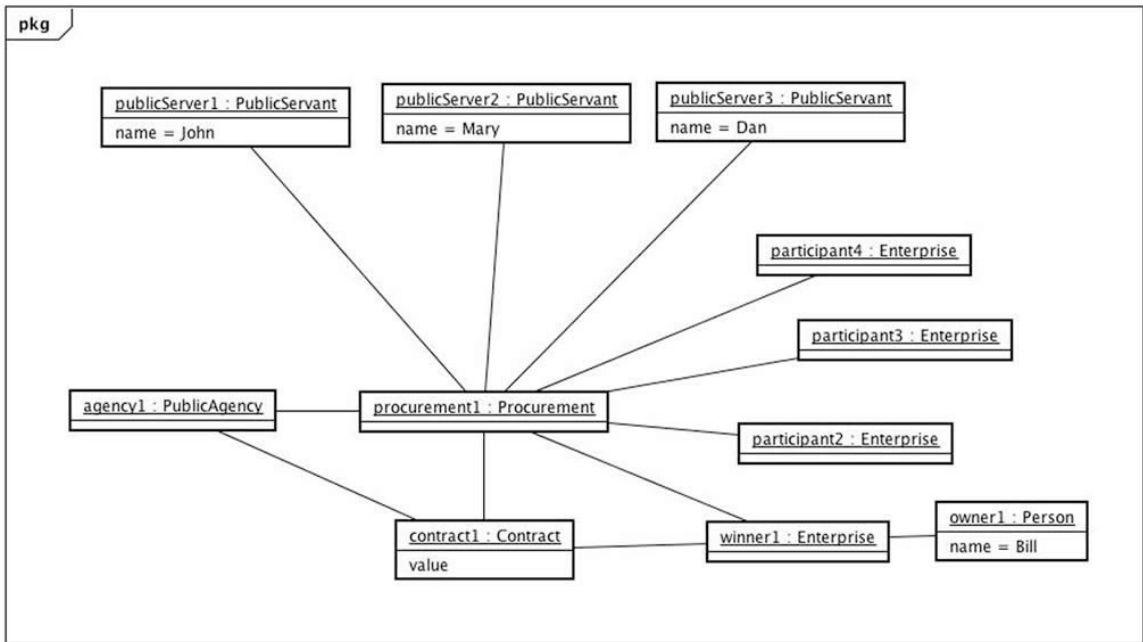


Fig. 2. A class diagram with instances of classes from the procurement domain

More complex modeling such as cardinalities of the attributes, disjoint and exhaustive knowledge, and formal axioms can be represented in UML only with the use of OCL. However, according to [Gómez-Pérez et al., 2005], there is no standard support for this language in common CASE tools. Because of this, and because UML models lack formal semantics, expressions in OCL cannot be evaluated by many CASE tools, and cannot be shared among developers.

In ER, with the common extension of generalization relationships between entities, it is possible to represent classes through the use of ER-entities. Furthermore, classes can be organized in taxonomies with the generalization relationship between ER-entities. For example, Figure 3 shows the class *PublicServant*, which is a subclass of the class *Person*.

It is also possible to represent attributes and their types through ER-attributes. In Figure 3 the class *Person* has the attributes *name* and *ID* with types *VARCHAR(255)* and *INTEGER*<sup>2</sup> respectively.

<sup>2</sup> In E/R, instead of using String and int, the types used are VARCHAR(size) and INTEGER, respectively.



Ad hoc relations can be represented through ER-relations between ER-entities. These relations can have any arity and can have specified cardinalities. Figure 3 presents several relations. One of these is *Apply*, which relates one or more *ScoreRule* to just one *Procurement*. Although a ternary relation can be easily represented, none were included in Figure 3 for reasons of clarity.

Instances can be created through the use of the *insert* sentence from the Structured Query Language (SQL), which essentially becomes a filled row in its respective table.

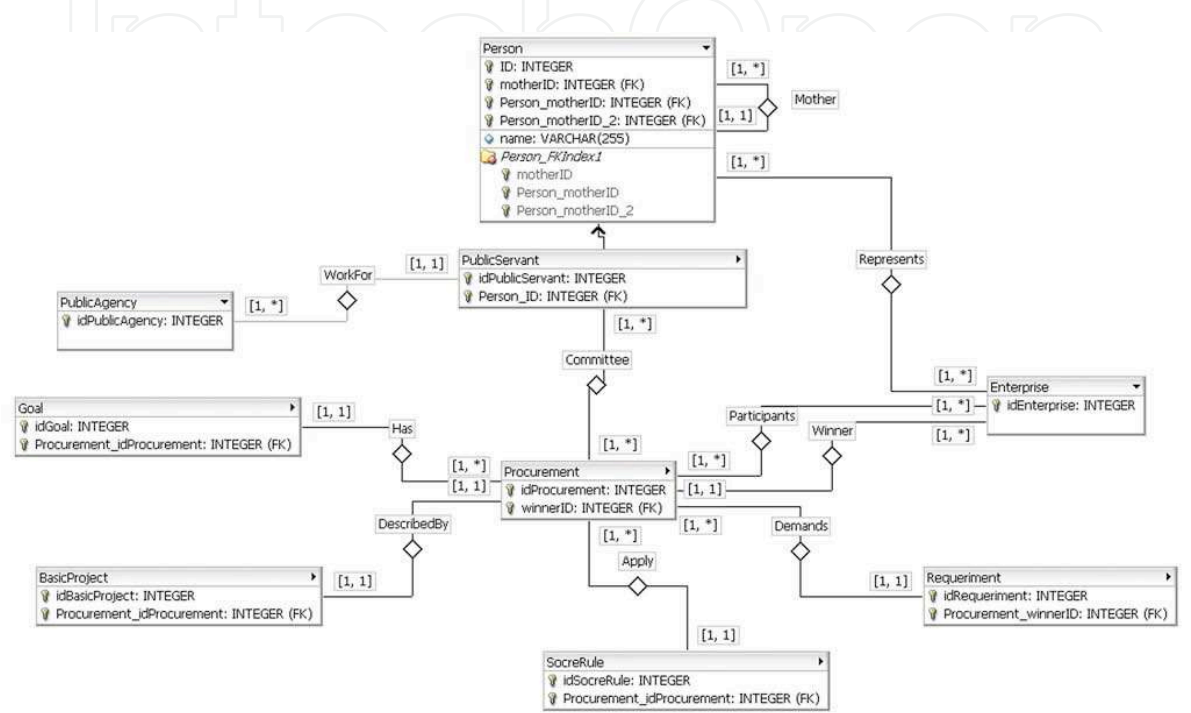


Fig. 3. An extended entity-relationship diagram for the procurement domain

According to [Gómez-Pérez et al., 2005], representing formal axioms in ER requires either extending the language, or using complementary notations, such as first-order logic or production rules.

3. Knowledge Representation and Reasoning

We have seen in Section 2 that UML and ER are able to represent classes, attributes, relations, and instances. However, they fall short when dealing with formal axioms. Knowledge representation systems, on the other hand, can naturally represent formal axioms. Furthermore, there are standard and readily available formal systems for reasoning with axioms.

According to [Levesque & Lakemeyer, 2000] “Knowledge Representation is the field of study within AI concerned with using formal symbols to represent a collection of propositions believed by some putative agent.” On knowledge-based systems, the information believed by this putative agent is usually explicitly represented by a collection of symbolic structures, in other words, represented on the knowledge base (KB). Reasoning, on the other hand, is the process of manipulating these symbols in order to produce new information that is not explicitly represented in the KB.

To see the benefits of using KR&R, we will augment the models presented in Section 2 by introducing axioms related to the domain. Based on Law N° 8,666/93, a member of a procurement committee must not be related to the enterprises that are participating in the procurement. For the sake of simplicity we will only deal with the relation that a public servant cannot be a brother, son/daughter, or mother of someone who is responsible for that enterprise. Consider the following simple representation:

```

01.  $\forall x$  PublicServant( $x$ )  $\Rightarrow$  Person( $x$ )
02.  $\forall x, y, z$  Mother( $x, y$ )  $\wedge$  Mother( $z, y$ )  $\Rightarrow$  Sibling( $x, z$ )
03.  $\forall x, y$  Mother( $x, y$ )  $\Rightarrow$  Related( $x, y$ )
04.  $\forall x, y$  Sibling( $x, y$ )  $\Rightarrow$  Related( $x, y$ )
05.  $\forall x, y, z, r$  Committee( $x, y$ )  $\wedge$  Participant( $z, y$ )  $\wedge$  Responsible( $r, z$ )
     $\Rightarrow \neg$ Related( $x, r$ )
06. PublicServant(John)
07. PublicServant(Mary)
08. PublicServer(Dan)
09. Person(Bill)
10. Person(Rebecca)
11. Mother(John, Rebecca)
12. Mother(Bill, Rebecca)
13. Procurement(Procurement1)
14. Committee(John, Procurement1)
15. Committee(Mary, Procurement1)
16. Committee(Dan, Procurement1)
17. Enterprise(Winner1)
18. Responsible(Bill, Winner1)
19. Participant(Winner1, Procurement1)
...

```

To be concise, we will not define the entire model. We assume the definitions are set up to reproduce the models from Section 2. Nevertheless, with the information presented above, we can identify an inconsistency. By combining lines 11, 12, 02, and 04, we can infer *Related(John, Bill)*. However, if we combine lines 14, 19, 18, and 05, we can infer that  $\neg$ *Related(John, Bill)*. Therefore, we have encountered an inconsistency in our KB. Fortunately, this representation allows us to debug, and to discover that this procurement is actually violating the law. We can then fix the inconsistency by modifying Rule 05 to say that if a member of a procurement committee is related to the enterprises that are participating in the procurement, then there is a violation of the law.

One could argue that this restriction can be easily implemented by adding an operation *isCommitteeRelatedToParticipants()* to the class *Procurement* from our UML model in Section 2, for instance. This operation would return true if there is a relation, as defined above, between one of the members of the committee and one of the responsible persons of the enterprises that participates in the procurement. However, UML lacks a formal way to define such an operation in detail, leaving its implementation open to the implementer. This has at least two main disadvantages. First, every system that uses this model has to implement its own interpretation of the operation. In addition to creating duplication of effort, this could easily lead to differing interpretations and inconsistent implementations. Second, if for some reason the rule changes (e.g., we realize we need to include father in our relation), then every interpretation of the model, i.e. every system that uses this model,



would have to change its implementation, instead of just changing the model as we would do in a knowledge-based system. This simple example illustrates several advantages of using KR&R in modeling.

One of the main advantages of knowledge-based systems is that it takes action based on what our putative agent believes, and not only on what is explicitly represented on the KB. We could easily see that in the example given above where the information inferred by the system was crucial in finding an infringement of the law.

In fact, the inference process described above is entailment, which is exactly what makes logic an important element of KR\&R, since logic is the study of entailment relations. According to \cite{levesque\_logic\_2001, brachman\_knowledge\_2004}, entailment is defined as:

*We say that the propositions represented by a set of sentences  $S$  entail the proposition represented by a sentence  $p$  when the truth of  $p$  is implicit in the truth of the sentences in  $S$ . In other words, entailment means that if the world is such that every element of  $S$  comes out true, then  $p$  does as well.*

According to [Levesque & Lakemeyer, 2000; Brachman & Levesque, 2004], a knowledge-based system can be seen as a system that performs some problem-solving activity. An example of such activity is verifying whether there is a member of the committee who is related to one of the responsible persons of an enterprise that participates in a specific procurement. It is able to do so by looking at what it already knows: Is a member mother of a responsible person of a participant enterprise? Does a member have the same mother as a responsible person of a participant enterprise? In order to answer such questions the system has to look at the explicit information available in its KB and make inferences based on it. Therefore, it is reasonable to separate the representation and management of the KB from the rest of the system. In other words, it is not important for a problem-solving system how the knowledge is store or how the reasoning is done to produce new knowledge. This kind of system is only concerned on answering its domain-specific questions, which in this case would be to answer questions related to fraud in public procurements.

According to [Levesque & Lakemeyer, 2000; Brachman & Levesque, 2004]:

*It is the role of a knowledge representation system to manage the KB within a larger knowledge-based system. Its job is to make various sorts of information about the world available to the rest of the system based on the information it has obtained, perhaps from other parts of the system, and by using whatever reasoning it can perform. Therefore, the job of the KR system is smaller than that of a full knowledge-based problem solver, but larger than that of a database management system, which would merely retrieve the contents of the KB.*

#### 4. Ontology and the Semantic Web

According to [Berners-Lee, 1999] the Semantic Web is a web of data that can be processed directly or indirectly by machines. This technology will drive us to a new phase where the arduous and manual task of identifying, accessing and utilizing information was successfully assigned to computers, allowing human beings to change their focus from data driven to knowledge driven activities.

The W3C [Heflin, 2004] states that ontologies are envisioned as the technology providing the cement for building the SW. Ontology was taken from Philosophy, where it means a systematic explanation of being. It contains a common set of terms for describing and representing a domain in a way that allows automated tools to use stored data in a wiser, context-aware fashion, intelligent software agents to afford better knowledge management, and many other possibilities brought by a standardized, more intensive use of metadata. [Studer et al., 1998] defines ontology as:

**Definition 1.** *An ontology is a formal, explicit specification of a shared conceptualization. Conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used, and the constraints on their use are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private of some individual, but accepted by a group.*

Since ontologies are used in different communities and with different objectives, [Uschold & Jasper, 1999] provided a new definition to make it more popular in all these different disciplines:

**Definition 2.** *An ontology may take a variety of forms, but it will necessarily include a vocabulary of terms and some specification of their meaning. This includes definitions and an indication of how concepts are inter-related which collectively impose a structure on the domain and constrain the possible interpretation of terms.*

According to [Gómez-Pérez et al., 2005] there are basically two distinct classes of ontologies: lightweight and heavyweight. The first describes the main concepts, its attributes and relationships while the second adds axioms that make the possible valid interpretations closer to its true intended meaning.

How do Ontology and the SW differ from what we have seen that UML, ER, and knowledge-based KR&R systems can model? Well, as seen before, the SW is designed for RIS environments, which are characterized by the AAA slogan, the open world assumption, and nonunique naming. But this style of information gathering can create a chaotic landscape rife with confusion, disagreement and conflict. UML, ER, and knowledge-based KR&R systems were developed under a more constrained paradigm for information sharing, and lack some important features needed to contain the chaos to which RIS environments are prone. A number of SW modeling languages have been developed expressly for the RIS environments. These languages differ in their capabilities and their level of expressivity, but all incorporate features necessary to foster cooperative and collaborative information sharing in RIS environments.

It is easy to see that our domain of fraud detection/prevention is a RIS environment. The data CGU has available does not come only from its audits and inspections. In fact, much complementary information can be retrieved from other Federal Agencies, including Federal Revenue Agency, Federal Police, and others. Imagine we have information about the enterprise that won the procurement, and we want to know information about its owners, such as their personal data and annual income. This type of information is not available at CGU's Data Base (DB), but must be retrieved from the Federal Revenue Agency's DB. Once the information about the owners is available, it might

be useful to check their criminal history. For that (see Figure 4), information from the Federal Police must be used. In this example, we have different sources saying different things about the same person: thus, the AAA slogan applies. Moreover, there might be other Agencies with crucial information related to our person of interest; in other words, we are operating in an open world. Finally, to make this sharing and integration process possible, we have to make sure we are talking about the same person, who may (especially in case of fraud) be known by different names in different contexts.



Fig. 4. Retrieving information through the SW for the procurement domain

5. Uncertainty in the Semantic Web

Consider our example from Section 3, in which we stipulate that a member of the procurement must not be related to a person responsible for an enterprise that is participating in the same procurement. Current SW deterministic reasoning algorithms will either consider this relation to be true, false, or unknown, with no way of expressing gradations of plausibility.

This is acceptable in situations where complete information is available. However, in open world environments such as the Web, partial (not complete) or approximate (not exact) information is more the rule than the exception. For example, we may not have the information from lines 11 and 12 from Section 3 stating that John and Bill have the same mother, Rebecca. However, we do have information about John and Bill stating that they have a common last name and live at the same address. Although we are uncertain about whether or how they are related, there is evidence suggesting they are. It is important to consider that information when reasoning about possible violations of procurement regulations.

Although the above and similar examples imply the need for principled representation and reasoning with uncertainty within the SW, current SW applications (including current automated reasoning methods) are primarily based on classical logic. This includes OWL, the W3C standard web ontology language, which has its logical basis in classical description logic, and therefore lacks built-in support for uncertainty. This is a major shortcoming for a technology intended to operate in RIS environments. The W3C responded to this limitation by initiating the Uncertainty Reasoning for the World Wide Web Incubator group (URW3-XG), created in 2007 and concluded a year later. The URW3-XG concluded that standardized representations were needed to express uncertainty in Web-based information [Laskey & Laskey, 2008].

Uncertainty is especially important to applications such as corruption prevention, in which perpetrators seek to conceal illicit intentions and activities. To address the SW lack of support to uncertainty, [Costa, 2005] proposed a Bayesian framework for probabilistic ontologies. Probabilistic ontologies have the expressiveness required for SW applications, and yet provide a principled logical basis for representing and reasoning under uncertainty. The probabilistic ontology language PR-OWL [Costa et al., 2005; Costa & Laskey, 2006] is based on Multi-Entity Bayesian Networks (MEBN) [Laskey, 2007; Laskey & Costa 2005] a probabilistic logic that combines the expressive power of First-Order Logic (FOL) with Bayesian networks' ability to perform plausible reasoning.

## 5.1 MEBN

Multi-Entity Bayesian Networks (MEBN) extend Bayesian Networks (BN) to achieve first-order expressive power. MEBN represents knowledge as a collection of MEBN Fragments (MFragments), which are organized into MEBN Theories (MTheories).

An MFragment contains random variables (RVs) and a fragment graph representing dependencies among these RVs. It represents a repeatable pattern of knowledge that can be instantiated many times as needed to form a BN addressing a specific situation, and thus can be seen as a template for building fragments of a Bayesian network. It is instantiated by binding its arguments to domain entity identifiers to create instances of its RVs. There are three kinds of RV: context, resident and input. Context RVs represent conditions that must be satisfied for the distributions represented in the MFragment to apply. Input RVs may influence the distributions of other RVs in an MFragment, but their distributions are defined in their home MFragments. Distributions for resident RV instances are defined within the MFragment by specifying local distributions conditioned on the values of the instances of their parents in the fragment graph.

A set of MFragments represents a joint distribution over instances of its random variables. MEBN provides a compact way to represent repeated structures in a BN. An important advantage of MEBN is that there is no fixed limit on the number of RV instances, and the random variable instances are dynamically instantiated as needed.

An MTheory is a set of MFragments that collectively satisfy conditions of consistency ensuring the existence of a unique joint probability distribution over its random variable instances.

To apply an MTheory to reason about particular scenarios, one needs to provide the system with specific information about the individual entity instances involved in the scenario. Upon receipt of this information, Bayesian inference can be used both to answer specific questions of interest (e.g., how likely is it that a particular procurement is being directed to a specific enterprise?) and to refine the MTheory (e.g., each new situation includes additional data about the likelihood of fraud for that set of circumstances).



Bayesian inference is used to perform both problem specific inference and learning from data in a sound, logically coherent manner.

## 5.2 PR-OWL

The usual workaround for representing uncertainty in languages lacking inbuilt support for it is to use custom-designed XML tags to annotate statements with numerical probabilities. This is a palliative solution that cannot represent how uncertainties depend on structural features of the domain. This is a major shortcoming that makes this simple approach unsuitable for all but the simplest real world problems involving uncertainty representation and reasoning. Researchers have consistently stressed the importance of structural information in probabilistic models (see [Schum, 1994]). For instance, [Shafer, 1986] stated that probability is more about structure than it is about numbers.

PR-OWL is a language for representing probabilistic ontologies. Probabilistic ontologies go beyond simply annotating ontologies with probabilities to provide a means of expressing subtle features required to express a first-order Bayesian theory. Because PR-OWL is based on MEBN logic, it not only provides a consistent representation of uncertain knowledge that can be reused by different probabilistic systems, but also allows applications to perform plausible reasoning with that knowledge, in an efficient way. Work on PR-OWL is based on the following definition of a probabilistic ontology [Costa, 2005]:

**Definition 3.** *A probabilistic ontology is an explicit, formal knowledge representation that expresses knowledge about a domain of application. This includes:*

- a. Types of entities existing in the domain;*
- b. Properties of those entities;*
- c. Relationships among entities;*
- d. Processes and events that happen with those entities;*
- e. Statistical regularities that characterize the domain;*
- f. Inconclusive, ambiguous, incomplete, unreliable, and dissonant knowledge;*
- g. Uncertainty about all the above forms of knowledge;*

*where the term entity refers to any concept (real or fictitious, concrete or abstract) that can be described and reasoned about within the domain of application.*

Probabilistic ontologies are used for the purpose of comprehensively describing knowledge about a domain and the uncertainty associated with that knowledge in a principled, structured, and sharable way. PR-OWL was developed as an extension enabling OWL ontologies to represent complex Bayesian probabilistic models in a way that is flexible enough to be used by diverse Bayesian probabilistic tools based on different probabilistic technologies (e.g. PRMs, BNs, etc). More specifically, PR-OWL is an upper ontology (i.e. an ontology that represents fundamental concepts that cross disciplines and applications) for probabilistic systems. PR-OWL is expressive enough to represent even the most complex probabilistic models. It consists of a set of classes, subclasses and properties that collectively form a framework for building probabilistic ontologies.

OWL has three different versions with increasing expressive power designed for specific communities of developers and users. The least expressive version is OWL Lite, which has a limited set of simple restrictions. The next step in expressiveness in the OWL family is OWL DL, which is based on Descriptive Logic and aims to maximize expressiveness while maintaining completeness (all logical consequences are provable) and

decidability (all proofs terminate in finite time). OWL-DL has all OWL constructions, but there are certain restrictions on use. The most expressive version, OWL Full, was built for users who want the strongest representational power possible in OWL format. As a consequence, there are no guaranties of computability. Following the same reasoning, a PR-OWL Lite version could be created as suggested in [Costa, 2005] with some restrictions.

PR-OWL was proposed as an extension to the OWL language based on MEBN, which can express a probability distribution on interpretations of any first-order theory. As a consequence, there are no guaranties that reasoning with PR-OWL ontology will be efficient or even decidable [Costa, 2005]. For problems in which computational efficiency is a concern, well-known classes of computationally efficient Bayesian theories can be represented in PR-OWL. PR-OWL was built to be interoperable with non-probabilistic ontologies. Since PR-OWL adds new definitions to OWL while retaining backward compatibility with its base language, OWL-built legacy ontologies will be able to interoperate with newly developed probabilistic ontologies. However, the ontology's probabilistic definitions have to form a valid complete or partial MTheory. Figure 5 shows the main concepts involved in defining an MTheory in PR-OWL.

In the diagram, ellipses represent general classes while arrows represent the main relationships between these classes. A probabilistic ontology (PO) has to have at least one individual of class MTheory, which is basically a label linking a group of MFrag that collectively form a valid MTheory. In actual PR-OWL syntax, that link is expressed via the object property *hasMFrag* (which is the inverse of object property *isMFragIn*). Individuals of class *MFrag* are comprised of nodes (not shown in the picture). Each individual of class *Node* is a random variable (RV) and thus has a mutually exclusive, collectively exhaustive set of possible states. In PR-OWL, the object property *hasPossibleValues* links each node with its possible states, which are individuals of class *Entity*. Finally, random variables (represented by the class *Node* in PR-OWL) have unconditional or conditional probability distributions, which are represented by class *ProbabilityDistribution* and linked to their respective nodes via the object property *hasProbDist*.

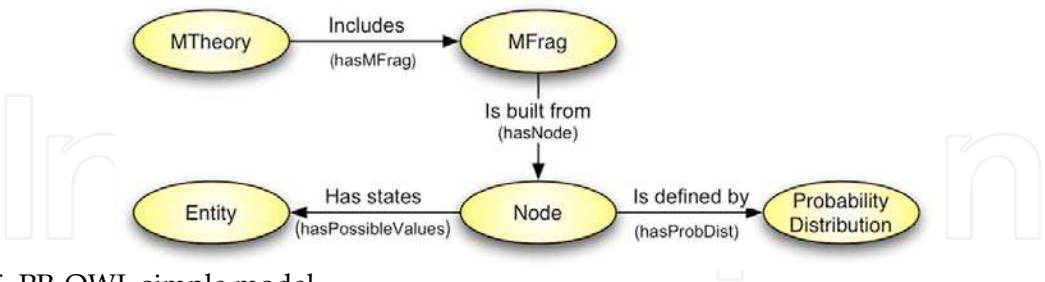


Fig. 5. PR-OWL simple model

Figure 6 depicts the main elements of the PR-OWL language, its subclasses, and the secondary elements necessary for representing an MTheory. The relations necessary to express the complex structure of MEBN probabilistic models using the OWL syntax are also depicted.



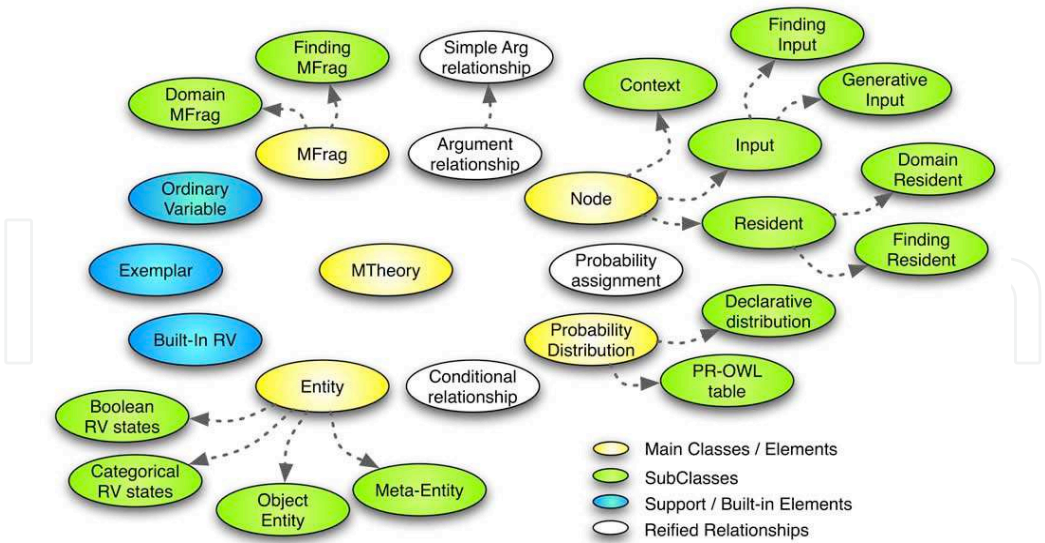


Fig. 6. PR-OWL detailed model

Previously, the first step towards building a probabilistic ontology as defined above is to import the PR-OWL ontology into an ontology editor (e.g. OntoEdit, Protégé, Swoop, etc.) and start constructing the domain-specific concepts using the PR-OWL definitions to represent uncertainty about their attributes and relationships. Using this procedure, a knowledge engineer is not only able to build a coherent generative MTheory and other probabilistic ontology elements, but also make it compatible with other ontologies that use PR-OWL concepts. However, building MFrag this way is a manual, error prone, and tedious process that requires deep knowledge of the logic and of the data structures of PR-OWL in order to avoid errors or inconsistencies. UnBBayes changes all that by providing a GUI-based editing process for building probabilistic ontologies based on the PR-OWL upper ontology for probabilistic theories [Carvalho et al., 2007a]. Another important feature is the ability to save and open models created by the UnBBayes GUI in PR-OWL format, with backwards compatibility to OWL through the use of the Protégé API. Protégé is an ontology editor and a flexible and configurable framework for building knowledge-based tools and applications. Protégé was developed by the Stanford Center for Biomedical Medical Informatics Research.

The major advantages of using PR-OWL are its flexibility and representational power, both inherited from the fact that the language is based on MEBN, a full integration of First-Order Logic (FOL) and probability theory that merges the expressiveness of the former with the inferential power of the latter. UnBBayes leverages this power with a built-in MEBN reasoner. The next section provides an overall view of the current state of that tool. The prospective reader can find additional details on PR-OWL at <http://www.pr-owl.org>.

6. Modeling a Probabilistic Ontology in UnBBayes

PR-OWL has been proposed as a language for representing uncertainty in the SW. Although there is now substantial literature about what PR-OWL is [Costa, 2005; Costa et al., 2005; Costa et al., 2008b], how to implement it [Carvalho et al., 2007; Carvalho et al., 2008a; Carvalho et al., 2008b; Costa et al., 2008a], and where it can be used [Costa et al., 2006; Costa

et al., 2009a; Costa et al., 2009b; Laskey et al., 2007; Laskey et al., 2008a; Laskey et al., 2008b;], little has been written about how to model a probabilistic ontology.

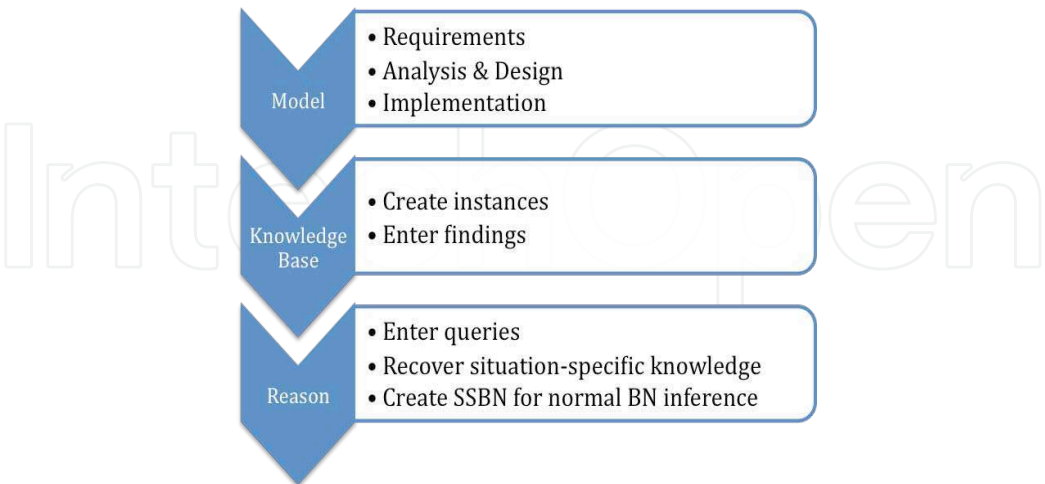


Fig. 7. Uncertainty Modeling Process for the SW (UMP-SW) using MEBN

Therefore, in this Section we will describe an approach for modeling a probabilistic ontology and how to use it for plausible reasoning in the SW. The Uncertainty Modeling Process for the SW (UMP-SW) presented in Figure 7 is divided into three steps: First we have to model the domain, then we need to populate the KB, and finally we can use the model and KB for reasoning.

The modeling step consists of three major stages: Requirements, Analysis & Design, and Implementation. These terms are borrowed from the Unified Process (UP) [Jacobson et al., 1999] with some modifications to reflect our domain of ontology modeling instead of development process. The methodology described here is also consistent with the Bayesian network modeling methodology described by [Laskey & Mahoney, 2000] and [Korb & Nicholson, 2003]. Figure 8 depicts these three stages of the Probabilistic Ontology Modeling Cycle (POMC). Like the UP, POMC is iterative and incremental. The basic idea behind iterative enhancement is to model our domain incrementally, allowing the modeler to take advantage of what was being learned during the modeling of earlier, incremental, deliverable versions of the model. Learning comes from discovering new rules, entities, and relations that were not obvious previously, which can give rise to new questions and evidence that might help us achieve our previously defined goal as well as give rise to new goals.

In the POMC (Figure 8) the Requirements stage (blue circle) defines the goals that must be achieved by reasoning with the semantics provided by our model. The Analysis and Design stage describes classes of entities, their attributes, how they relate, and what rules apply to them in our domain (green circles). This definition is independent of the language used to implement the model. Finally, the Implementation stage maps our design to a specific language that allows uncertainty in the SW, which in this case is PR-OWL (red circles).

We will now illustrate the POMC through a case study in procurement fraud detection and prevention. We also demonstrate the use of UnBBayes to implement the model, to populate the KB, and to perform plausible reasoning.



Fig. 8. Probabilistic Ontology Modeling Cycle (POMC) - Requirements in blue, Analysis & Design in green, and Implementation in red

6.1 Requirements

The objective of the requirements stage is to define the objectives that must be achieved by representing and reasoning with a computable representation of domain semantics. At this stage, it is important to define the questions that the model is expected to answer (i.e., the queries to be posed to the system being designed). For each question, a set of information that might help answer such question (evidence) must be defined.

In order to understand the requirements for the procurement fraud detection and prevention model, we first have to explain some of the problems encountered when dealing with public procurements.

One of the principles established by the Law N° 8,666/93 is equality among the bidders. This principle prohibits the procurement agent from discriminating among potential suppliers. However, if the procurement agent is related to the bidder, he/she might feed information or define new requirements for the procurement in a way that favors the bidder.

Another principle that must be followed in public procurement is that of competition. Every public procurement should establish minimum requisites necessary to guarantee the execution of the contract in order to maximize the number of participating bidders. Nevertheless, it is common to have a fake competition when different bidders are, in fact, owned by the same person. This is usually done by having someone as a front for the

enterprise, which is often someone with little or no education. Another common tactic is to set up front enterprises owned by relatives of the enterprise committing fraud.

According to [Mueller, 1998] participating in a public procurement can be very expensive and time consuming. Thus, some firms are unwilling to take part in a process that is not guaranteed to achieve favorable results. Since this diminishes the number of enterprises participating in the procurement, collusion among the bidders is more likely to happen. What happens in Brazil is that a small group of firms regularly participate in procurements of certain goods and services. When this happens, the competitors in a public procurement take turns winning the contracts. They stipulate the winning bid, and all other firms bid below that price. There is no competition, and the government pays a higher price for the contract. Although collusion is not an easy thing to prove, it is reasonable to assume that collusion is enabled by some kind of relationship between the enterprises.

All firms in Brazil have a registration number, called CGC, which stands for General List of Contributors. When a firm is suspended from procuring with the public administration, its CGC number is used to inform all other public agencies that this firm should not participate in public procurements. However, the firm can simply close its business and open a new one using a different CGC. Thus the firm that should not be able to participate in public procurements is now allowed, since it now has a different number associated to it. Unfortunately, the Commercial Code permits this change of CGC number.

One other problem is that public procurement is quite complex and may involve large sums of money. Therefore, the members that form the committee of the procurement must not only be prepared, but also have a clean history (no criminal nor administrative conviction) in order to maximize morality, one of the ethical principles that federal, state, municipal and district government should all adopt.

Having explained that, in our fraud detection and prevention in the procurements domain we have the following set of goals/queries/evidences:

1. Identify if a given procurement should be inspected and/or audited (i.e. evidence suggests further analysis is needed);
  - a. Is there any relation between the committee and the enterprises that participated in the procurement?
    - i. Look for member and responsible person of an enterprise who are related (mother, father, brother, or sister);
    - ii. Look for member and responsible person of an enterprise who live at the same address.
  - b. Is the responsible person of the winner enterprise of the procurement a front?
    - i. Look for value of the contract related to this procurement;
    - ii. Look for his education degree;
    - iii. Look for annual income.
  - c. Was the responsible person of the winner enterprise of the procurement responsible for an enterprise that has been suspended from procuring with the public administration?
    - i. Look for this information in the General List of Contributors (CGC) database.
  - d. Was competition compromised?
    - i. Look for bidders related as defined above (1a).
2. Identify whether the committee of a given procurement should be changed.
  - a. Is there any member of committee who does not have a clean history?

- i. Look for criminal history;
  - ii. Look for administrative investigations.
- b. Is there any relation between members of the committee and the enterprises that participated in previous procurements?
  - i. Look for member and responsible person of an enterprise who are relatives (mother, father, brother, or sister);
  - ii. Look for member and responsible person of an enterprise who live at the same address.

## 6.2 Analysis & Design

Once we have defined our goals and described how to achieve them, it is time to start modeling the entities, their attributes, relationships, and rules to make that happen. This is the purpose of the Analysis and Design stage.

The major objective of this stage is to define the semantics of our model. In fact, most of our semantics can be defined in normal ontologies, including the deterministic rules that the concepts described in our model must obey. Since there are whole books describing how to design such ontologies, and our main concern is on the uncertain part of the ontology, we will not cover these methods in this Section. For more information see [Allemang & Hendler, 2008; Gómez-Pérez et al., 2005].

Nevertheless, we do need a starting point in order to design our probabilistic ontology. As a matter of fact, one good way to start modeling these properties is to use UML as described in Section 2. However, as we have seen, UML does not support complex rule definitions. So we will just document them separately to remind us of the rules that must be described when implementing our model in PR-OWL.

Figure 9 depicts a simplified design of our domain requirements. A *Person* has a *name*, a *mother* and a *father*<sup>3</sup> (also *Person*). Every *Person* has a unique identification that in Brazil is called *CPF*. A *Person* also has an *Education* and lives at a certain *Address*. In addition, everyone is obliged to file his/her *TaxInfo* every year, including his/her *annualIncome*. These entities can be grouped as *Personal Information*. A *PublicServant* is a *Person* who works for a *PublicAgency*, which is a *Government Agency*. Every public *Procurement* is owed by a *PublicAgency*, has a *committee* formed by a group of *PublicServants*, and has a group of *participants*, which are *Enterprises*. One of these will be the *winner* of the *Procurement*. Eventually, the *winner* of the *Procurement* will receive a *Contract* of some *value* with the *PublicAgency* owner of the *Procurement*. The entities just described can be grouped as *Procurement Information*. Every *Enterprise* has at least one *Person* that is *responsible* for its legal acts. An *Enterprise* also has an identification number, the General List of Contributors *CGC*, which can be used to inform that this *Enterprise* is suspended from procuring with the public administration, *isSuspended*. These are grouped as the *Enterprise Information*. We also have *AdministrativeInvestigation*, which has information about investigations that *involves* one or more *PublicServer*. Its *finalReport*, the *JudgmentAdministrativeReport*, contains information about the penalty applied, if any. These entities form the *Administrative Judgment Information*. Finally we have the *Criminal Judgment Information* group that describes the *CriminalInvestigation* that *involves* a *Person*, with its *finalReport*, the *JudgmentCriminalReport*, which has information about the verdict.

<sup>3</sup> When modeling in UML we will use the first letter uppercase for classes (e.g., *Person*) and lowercase for attributes (e.g., *name*).



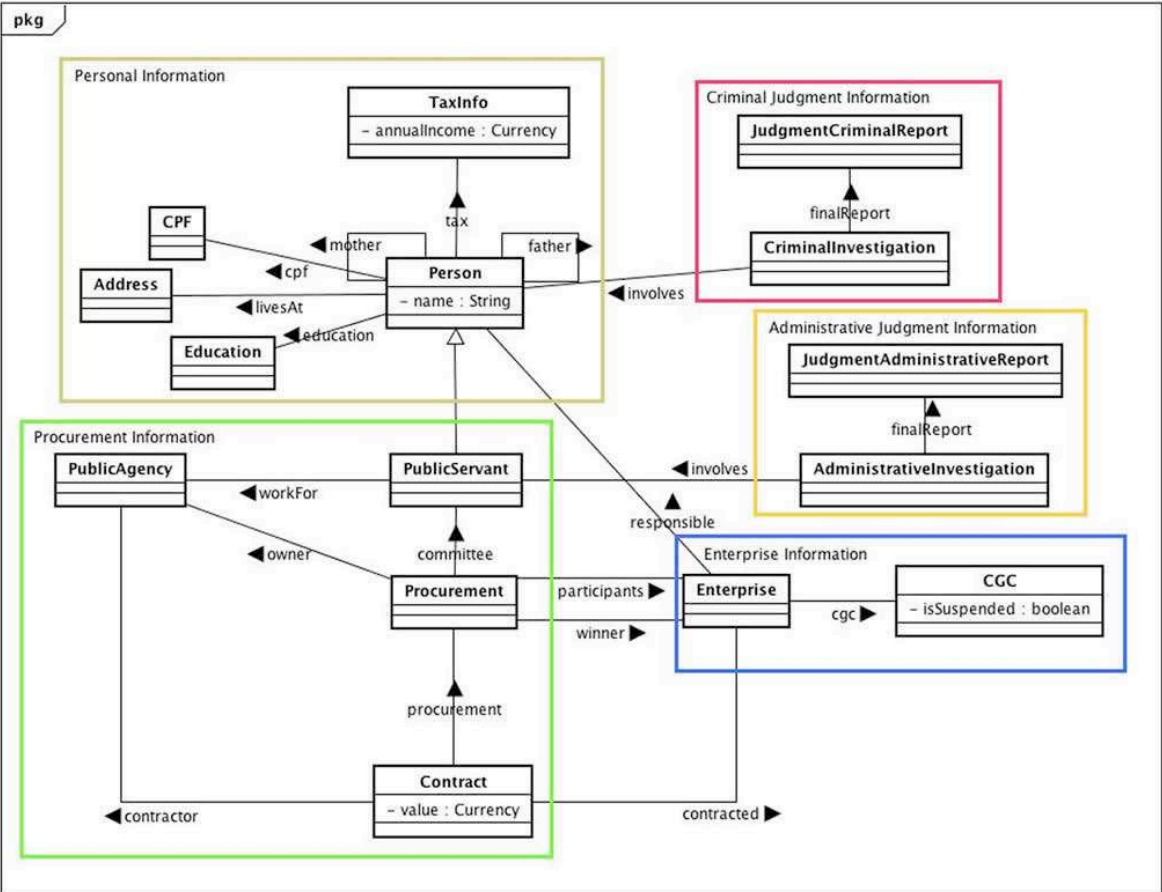


Fig. 9. Entities, their attributes, and relations for the procurement model

Besides the cardinality and uniqueness rules defined in the explanation above about the entities depicted in Figure 9, the probabilistic rules for our model include:

1. If a member of the committee has a relative (mother, father, brother, or sister) responsible for a bidder in the procurement, then it is more likely to exist a relation between the committee and the enterprises, which inhibits competition.
2. If a member of the committee lives at the same address as a person responsible for a bidder in the procurement, then it is more likely to exist a relation between the committee and the enterprises, which lowers competition.
3. If a contract of high value related to a procurement has a responsible person of the winner enterprise with low education or low annual income, then this person is likely to be a front for the firm, which lowers competition.
4. If the responsible person of the winner enterprise is also responsible for another enterprise that has its CGC suspended for procuring with the public administration, then this procurement is more likely to need further investigation.
5. If the responsible people for the bidders in the procurement are related to each other, then a competition is more likely to have been compromised.



- 6. If 1, 2, 3, or 5, then the procurement is more likely to require further investigation.
- 7. If a member of the committee has been convicted of criminal crime or has been penalized administratively, then he/she does not have a clean history. If he/she was recently investigated, then it is likely that he/she does not have a clean history.
- 8. If the relation defined in 1 and 2 is found in previous procurements, then it is more likely that there will be a relation between this committee and future bidders.
- 9. If 7 or 8, then it is more likely that the committee needs to be changed.

6.3 Implementation

Once we have finished our Analysis and Design, it is time to start implementing our model in a specific language. This Chapter describes how to model procurement fraud detection and prevention in PR-OWL using UnBBayes. The first thing to do is to start mapping the entities, their attributes, and relations to PR-OWL, which uses essentially MEBN terms. It is often a good idea to start mapping the entities. There is no need to map all entities in our model to an entity in PR-OWL. In fact, in our model we will make many simplifications. One of them is due to a limitation in UnBBayes’ current version, which is the lack of support for a type hierarchy. Therefore, we will not have the *PublicServant* entity and we will assume that a *Person* might work for a *PublicAgency*. We will also assume that every *Person* and *Enterprise* in our KB is uniquely identified by its name, so we will not consider, in this simplified example, the *CPF* and *CGC* entities. Figure 10 presents the entities implemented in our PR-OWL ontology using UnBBayes. For more details about defining entities in UnBBayes see [Carvalho et al., 2009].

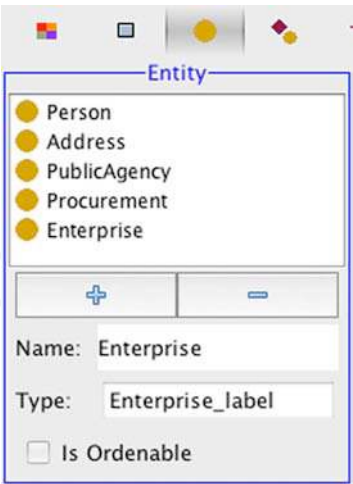


Fig. 10. Entities implemented in PR-OWL using UnBBayes

Once we have our entities defined, we consider characteristics that may be uncertain. Uncertainty is represented in MEBN by defining random variables (RVs). To define a RV in UnBBayes, we first define its Home MFrag. Grouping the RVs into MFrag is done by examining the grouping created during the Analysis and Design stage.

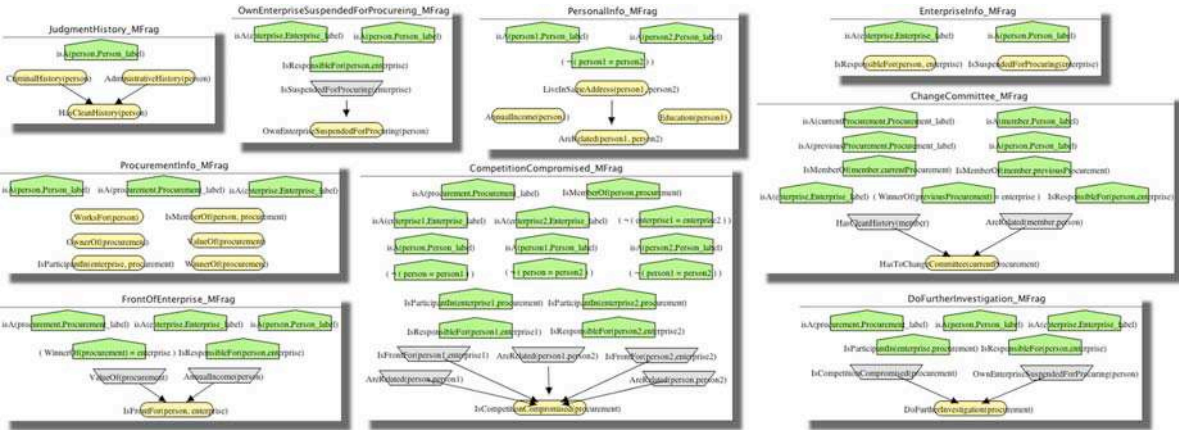


Fig. 11. Probabilistic ontology for fraud detection and prevention in the procurements

Typically, a RV represents an attribute or a relation from our designed model. For instance, the RV *LivesAt(Person)* maps the relation *livesAt* in our designed model. As it is a functional relation, *livesAt* relates a *Person* to an *Address*. Hence, the possible values (or states) of this RV are instances of *Address*. We can also avoid explicitly representing some entities, by simply defining discrete outputs. In our implementation, we only need to know if a *Person* has no education, just high school, an undergraduate degree, or a graduate degree. These are the states of the RV *Education(Person)*, therefore, there is no need to define the entity *Education*. Each of these RVs is represented in UnBBayes as a resident node in its home MFragment.

Because the current version of UnBBayes does not support continuous RVs, we must define a discretization for numerical attributes. For example, the attribute *value* of the *Contract* entity from our designed model is continuous, since it represents a *Currency*. However, we can discretize it by defining common intervals, as lower than 10,000.00, between 10,000.01 and 100,000.00, between 100,000.01 and 500,000.00, between 500,000.01 and 1,000,000.00, and greater than 1,000,000.01, which will be the sates of the resident node *ValueOf(Procurement)*. Future versions of UnBBayes will support continuous RVs.

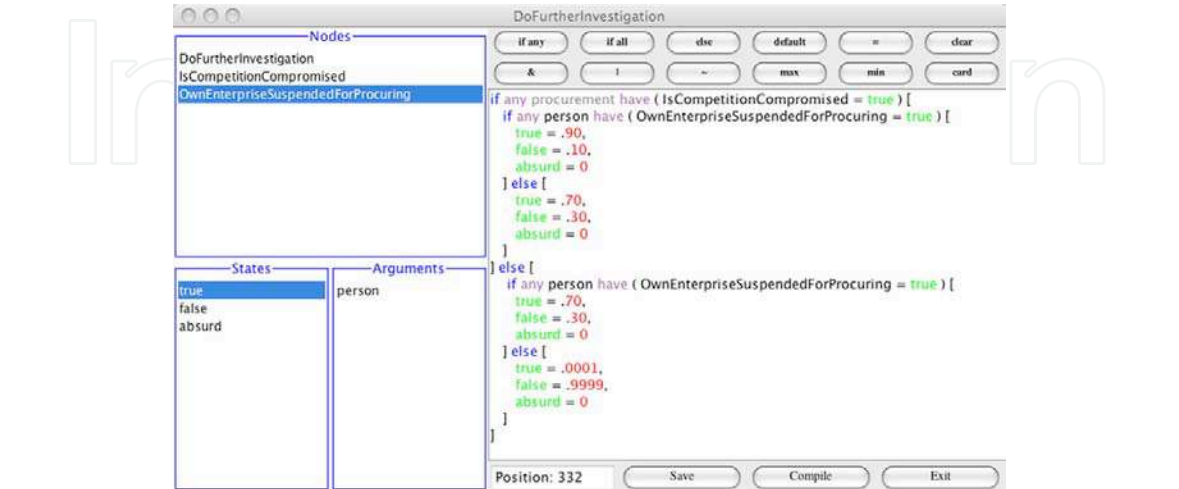


Fig. 12. Local probability distribution for node *DoFurtherInvestigation(Procurement)*

Once all resident RVs are created, their relations can be defined by analyzing dependence between nodes. One good way to look for dependence is by looking at the rules defined in our model. For instance, rule 3 indicates that there is a dependence between *ValueOf(Procurement)*, *Education(Person)*, and *IsFront(Person,Enterprise)*.

Figure 11 presents an MTheory that represents the final probabilistic ontology for the procurement fraud detection and prevention model. This MTheory is composed of nine MFrag. In each MFrag, the resident RVs are shown as yellow ovals; the input RVs are shown as gray trapezoids; the context RVs are shown as green pentagons. The two main goals described in our requirements are defined in the *DoFurtherInvestigation\_MFrag* and *ChangeCommittee\_MFrag*. A more sophisticated design to model whether to change the committee would define a utility function and use expected utility to make the decision. Future versions of UnBBayes will support Multi-Entity Influence Diagrams [Costa, 2005].

The final step in constructing a probabilistic ontology in UnBBayes is to define the local probability distribution (LPD) for all resident nodes. Figure 12 presents a LPD for the resident node *DoFurtherInvestigation(Procurement)*, which represents one of the main goals we want to achieve in our model. This LPD follows a grammar defined to allow declaring a more robust distribution. For more information see [Carvalho et al., 2008b].

6.4 Knowledge Base

After defining a probabilistic ontology, we can populate the KB by adding instances and findings. These can be added manually through or retrieved from a database (DB). Figure 13 shows how to add entities and findings manually using UnBBayes' GUI.

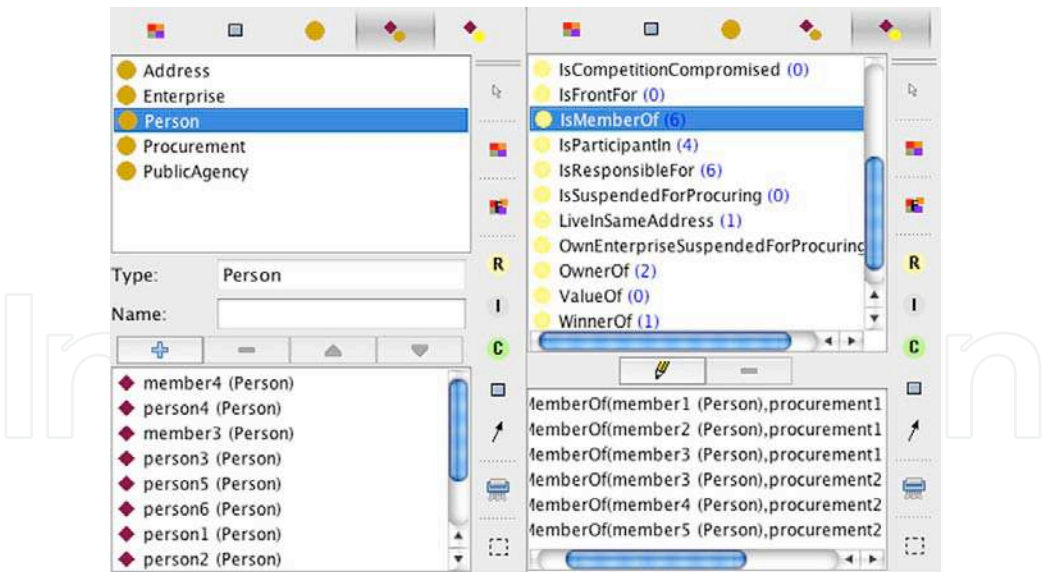


Fig. 13. Entities and findings for a specific situation in our probabilistic ontology

This specific situation has two procurements. The first, *procurement1*, had *member1*, *member2*, and *member3* forming its committee. This procurement was realized in the past and had *enterprise3* as its winner. Today, *procurement2* is still in progress and it has *member3*, *member4*, and *member5* forming its committee. It is also known that *member3* lives at the same address as *person3*, who is responsible for *enterprise3*, and that *member5* has been convicted by an administrative judgment.

6.5 Reasoning

Once we have a probabilistic ontology implemented in PR-OWL and we have a KB populated, it is possible to realize plausible reasoning through the process of creating a Situation-Specific Bayesian Network (SSBN) [Laskey, 2008]. UnBBayes has implemented an algorithm that creates a SSBN for a particular query (see [Carvalho et al., 2008a; Carvalho et al., 2008b; Costa et al., 2008a] for more details). In this specific situation we want to know if the current *procurement2* should change its committee, query *HasToChangeCommittee(procurement2)*. Figure 14 shows the generated SSBN.

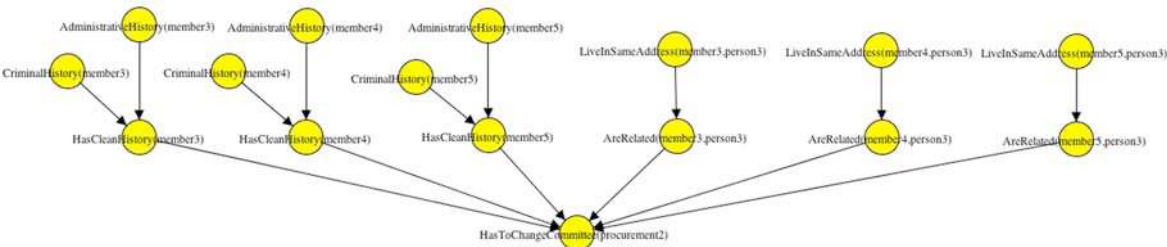


Fig. 14. SSBN for the query *HasToChangeCommittee(procurement2)*

With the current KB, the probability that *HasToChangeCommittee(procurement2)* is 87.76%. If we did not know that *member3* lives at the same address as *person3*, the probability would be 69.38%. On the other hand, if we did not know that *member5* has been convicted by an administrative judgment, the probability would be 63.84%. Finally, if we had no knowledge about neither of the information stated above, the probability would be only 2.6%.

7. Conclusion

Ontologies provide the “semantic glue” to enable knowledge sharing among systems collaborating in radical information sharing domains: open world domains in which Anyone can say Anything about Any topic (AAA), and entities may not have unique names. Traditional ontologies fail to provide adequate support for uncertainty, a ubiquitous characteristic of RIS environments. This paper presents a framework for modeling RIS domains using probabilistic ontologies. The case study presented in this work has shown that such research, albeit in its infancy, can help to support interoperability among systems in an open environment, fusing multiple sources of noisy information to perform reasoning and problem solving in an open world.

8. References

Allemang, Dean & Hendler, James A. (2008) *Semantic web for the working ontologist modeling in RDF, RDFS and OWL*. Elsevier, ISBN 978-0-12-373556-0, United States.

Berners-Lee, Tim (1999) *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*. Collins Business, ISBN 978- 0062515872.

Brachman, Ronald J. & Levesque, Hector J. (2004) *Knowledge representation and reasoning*. Elsevier, ISBN 978-1-55860-932-7, United States.



- Carvalho, Rommel N.; Santos, Laécio L.; Ladeira, Marcelo & Costa, Paulo C. G. (2007) A GUI Tool for Plausible Reasoning in the Semantic Web using MEBN. *In Proceedings of the Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007)*. Mourele, L.; Nedjah, N.; Kacprzyk, J.; and Abraham, A. (eds.); pp. 381-386. October 22-24, 2007, Rio de Janeiro, Brazil.
- Carvalho, Rommel N.; Ladeira, Marcelo; Santos, Laécio L.; Matsumoto, Shou & Costa, Paulo C. G. (2008a) A GUI Tool for Plausible Reasoning in the Semantic Web Using MEBN. *Chapter in Innovative Applications in Data Mining*. Studies in Computational Science, Vol. 169/2009, pp. 17-45. Springer-Verlag: Berlin/Heidelberg.
- Carvalho, Rommel N.; Ladeira, Marcelo; Santos, Laécio L.; Matsumoto, Shou & Costa, Paulo C. G. (2008b) UnBBayes-MEBN: Comments on Implementing a Probabilistic Ontology Tool. *In Proceedings of the IADIS International Conference - Applied Computing 2008*. N. Guimarães and P. Isaías (eds.), Algarve, Portugal, April 10-13, 2008, pp. 211-218.
- Chen, Peter PS. (1976) The Entity-Relationship Model: Toward a Unified View of Data. *In ACM Transactions on Database Systems (TODS)*, Volume 1, Issue 1, Framingham, MA, pp. 9-36.
- Costa, Paulo C. G. (2005) Bayesian Semantics for the Semantic Web. Doctoral Dissertation. Department of Systems Engineering and Operations Research, George Mason University: Fairfax, VA, USA. p. 312.
- Costa, Paulo C. G.; Laskey, Kathryn B. & Laskey, Kenneth J. (2005) PR-OWL: A Bayesian Framework for the Semantic Web. *In Proceedings of the first workshop on Uncertainty Reasoning for the Semantic Web (URSW 2005), held at the Fourth International Semantic Web Conference (ISWC 2005)*. November 6-10, 2005, Galway, Ireland.
- Costa, Paulo C. G.; Laskey, Kathryn B. & Laskey, Kenneth J. (2006) Probabilistic Ontologies for Efficient Resource Sharing in Semantic Web Services. *In Proceedings of the second workshop on Uncertainty Reasoning for the Semantic Web (URSW 2006), held at the Fifth International Semantic Web Conference (ISWC 2006)*. November 5-9, 2006, Athens, GA, USA.
- Costa, Paulo C. G.; Ladeira, Marcelo; Carvalho, Rommel N.; Laskey, Kathryn B.; Santos, Laécio L. & Matsumoto, Shou (2008a) A First-Order Bayesian Tool for Probabilistic Ontologies. *In Proceedings of the 21st International Florida Artificial Intelligence Research Society Conference (FLAIRS-21)*, Palo Alto: AAAI Press, 2008.
- Costa, Paulo C. G.; Laskey, Kathryn B. & Laskey, Kenneth J. (2008b) PR-OWL: a Bayesian Language for the Semantic Web. *Chapter in Uncertainty Reasoning for the Semantic Web I*. LNAI 5327, pp. 88-107. Springer-Verlag: Berlin/Heidelberg.
- Costa, Paulo C. G.; Laskey, Kathryn B. & Chang, KC (2009a) PROGNOS: Applying Probabilistic Ontologies to Distributed Predictive Situation Assessment in Naval Operations. *Accepted to the Fourteenth International Command and Control Research and Technology Conference (ICCRTS 2009)*. June, 15-17. Washington, D.C., USA.
- Costa, Paulo C. G.; Chang, KC; Laskey, Kathryn B. & Carvalho, Rommel N. (2009b) A Multi-Disciplinary Approach to High Level Fusion in Predictive Situational Awareness. *Accepted to the Eleventh International Conference of the Society of Information Fusion (FUSION 2009)*. July 6-9. Seattle, WA, USA.
- Crevelde, Martin V. *Command in War*. 1987 reprint edition. Harvard University Press Cambridge, ISBN 978-0674144415, MA, USA.

- Gómez-Pérez, Asunción; Fernández-López, Mariano; Corcho, Oscar (2005) *Ontological engineering: with examples from the areas of knowledge management, e-commerce and the semantic web*. Springer-Verlag, ISBN 1-85233-551-3, London Berlin Heidelberg.
- Heflin, Jeff (2004) OWL Web Ontology Language - Use Cases and Requirements (W3C Recommendation). [www.w3.org/TR/2004/REC-webont-req-20040210](http://www.w3.org/TR/2004/REC-webont-req-20040210).
- Jacobson, Ivar; Booch, Grady & Rumbaugh, James (1999) *The Unified Software Development Process*. Addison-Wesley Professional, ISBN 978-0201571691.
- Korb, Kevin B. & Nicholson, Ann E. (2003) *Bayesian Artificial Intelligence*. Chapman and Hall, Boca Raton.
- Laskey, Kathryn B. & Mahoney, Suzanne M. (2000) Network Engineering for Agile Belief Network Models. *IEEE Transactions in Knowledge and Data Engineering* 12(4): 487-498.
- Laskey, Kathryn B. & Costa, Paulo .C. G. (2005) Of Klingons and Starships: Bayesian Logic for the 23rd Century. In *Uncertainty in Artificial Intelligence: Proceedings of the Twenty-first Conference (UAI 2005)*. AUA Press: Edinburgh, Scotland.
- Laskey, Kathryn B.; Costa, Paulo C. G.; Wright, Edward J. & Laskey, Kenneth J. (2007) Probabilistic Ontology for Net-Centric Fusion. In *Proceedings of the Tenth International Conference of the Society of Information Fusion (FUSION 2007)*. July 9 - 12, 2007, Québec, Canada.
- Laskey, Kathryn B. (2008) MEBN: A language for first-order Bayesian knowledge bases. In *Artificial Intelligence*, Volume 172, Issues 2-3, February 2008, Pages 140-178
- Laskey, Kathryn B.; Costa, Paulo C. G. & Janssen, T. (2008a) Probabilistic Ontologies for Knowledge Fusion. In *Proceedings of the Eleventh International Conference of the Society of Information Fusion (FUSION 2008)*. July 1-3, 2008, Cologne, Germany.
- Laskey, Kathryn B.; Costa, Paulo C. G. & Janssen, T. (2008b) Probabilistic Ontologies for Multi-INT Fusion. In *Proceedings of the AFCEA-GMU C4I Center Symposium (GMU-AFCEA 2008) - "Critical Issues in C4I"*. May 20-21, 2008, George Mason University, Fairfax, VA, USA.
- Laskey, Kenneth & Laskey, Kathryn. Uncertainty reasoning for the world wide web: Report on the URW3-XG incubator group. URW3-XG, W3C, 2008.
- Levesque, Hector J. & Lakemeyer, Gerhard (2000) *The Logic of Knowledge Bases*. The MIT Press, ISBN 0-262-12232-4, Cambridge, Massachusetts and London, England.
- Meirelles, Hely L. (1996) *Licitação e Contrato Administrativo*. Malheiros Editores, Brazil.
- Mueller, André P. M. (1998) A Critical Study of the Brazilian Procurement Law. In *Minerva Research Papers*. The Institute of Brazilian Issues - IBI, The George Washington University, Washington DC.
- Object Management Group (1997) *Object Constraint Language Specification*. <http://www.omg.org/docs/ad/97-08-08.pdf>.
- Patel-Schneider, Peter F.; Hayes, Patrick & Horrocks, Ian (2004) OWL Web Ontology Language - Semantics and Abstract Syntax (W3C Recommendation). [www.w3.org/TR/owl-semantics/](http://www.w3.org/TR/owl-semantics/).
- Pearl, Judea (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, ISBN 978-1558604797, San Mateo, CA, USA.
- Rumbaugh, James; Jacobson, Ivar & Booch, Grady (1998) *The Unified Modeling Language Reference Manual*. Addison-Wesley, ISBN 978-0201309980, Boston, Massachusetts.
- Schum, David A. (1994) *Evidential Foundations of Probabilistic Reasoning*. Wiley- Interscience, ISBN 978-0471579366.



- Shafer, Glenn (1986) The Construction of Probability Arguments. *Boston University Law Review*, 66(3-4):799-823.
- Studer, Rudi; Benjamins, V. Richard & Fensel, Dieter (1998) Knowledge Engineering: Principles and Methods. In *IEEE Transactions on Data and Knowledge Engineering*, Volume 25, Numbers 1-2, pp. 161-197.
- Uschold, Mike & Jasper, Robert (1999) A Framework for Understanding and Classifying Ontology Applications. In *Benjamins BR (ed) IJCAI'99 Workshop on Ontology and Problem Solving Methods: Lessons Learned and Future Trends*. Stockholm, Sweden. CEUR Workshop Proceedings 18:11.1-11.12. Amsterdam, The Netherlands <http://CEUR-WS.org/Vol-18/>.
- Warner, Jos B. & Kleppe, Anneke G. (1998) *The Object Constraint Language: Precise Modeling With UML*. Addison-Wesley, ISBN 978-0201379402, Boston, Massachusetts.

IntechOpen



## **Semantic Web**

Edited by Gang Wu

ISBN 978-953-7619-54-1

Hard cover, 310 pages

**Publisher** InTech

**Published online** 01, January, 2010

**Published in print edition** January, 2010

Having lived with the World Wide Web for twenty years, surfing the Web becomes a way of our life that cannot be separated. From latest news, photo sharing, social activities, to research collaborations and even commercial activities and government affairs, almost all kinds of information are available and processible via the Web. While people are appreciating the great invention, the father of the Web, Sir Tim Berners-Lee, has started the plan for the next generation of the Web, the Semantic Web. Unlike the Web that was originally designed for reading, the Semantic Web aims at a more intelligent Web severing machines as well as people. The idea behind it is simple: machines can automatically process or “understand” the information, if explicit meanings are given to it. In this way, it facilitates sharing and reuse of data across applications, enterprises, and communities. According to the organisation of the book, the intended readers may come from two groups, i.e. those whose interests include Semantic Web and want to catch on the state-of-the-art research progress in this field; and those who urgently need or just intend to seek help from the Semantic Web. In this sense, readers are not limited to the computer science. Everyone is welcome to find their possible intersection of the Semantic Web.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Rommel Carvalho, Kathryn Laskey, Paulo Costa, Marcelo Ladeira, Laecio Santos and Shou Matsumoto (2010). UnBBayes: Modeling Uncertainty for Plausible Reasoning in the Semantic Web, Semantic Web, Gang Wu (Ed.), ISBN: 978-953-7619-54-1, InTech, Available from: <http://www.intechopen.com/books/semantic-web/unbbayes-modeling-uncertainty-for-plausible-reasoning-in-the-semantic-web>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元

[www.intechopen.com](http://www.intechopen.com)

Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

Phone: +86-21-62489820  
Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen