*Book Title*
*Book Editors*
*IOS Press, 2003*



# The Legal Concepts and the Layman's Terms

## *Bridging the Gap through Ontology-Based Reasoning about Liability*

Ronny van Laarschot [a], Wouter van Steenbergen [b], Heiner Stuckenschmidt [a], Arno R. Lodder [b] and Frank van Harmelen [a]

[a] *Department of Artificial Intelligence, Vrije Universiteit Amsterdam*
[b] *Computer/Law Institute, Vrije Universiteit Amsterdam*

**Abstract.** The aim of the BEST-project is to support laymen in judging their legal position through intelligent disclosure of case-law in the area of Dutch tort law. A problem we have to face in this context is the discrepancy between the terminology laymen use to describe their case and the terminology found in legal documents. We address this problem by supporting users to describe their case in common sense terms taken from an ontology. We use logical reasoning to automatically determine law articles that are relevant for determining liability of parties in a case based on this description, thus bridging the gap between the laymen's description and the terminology relevant for certain articles that can be found in legal documents. We introduce the BEST-project and describe the ontology built for supporting case descriptions focussing on its use for automatically determining relevant articles of law.

**Keywords.** Legal Ontologies, Logical Reasoning, Mediation, Document Retrieval

## 1. Introduction

Laymen can turn to legal professionals to determine their legal position, but often resolve their disputes in an informal way, e.g. by mediation, negotiation. In the BEST-project (Batna Establishment using Semantic web Technology, see http://best-project.nl) we strive to provide disputing parties with information about their legal position in a liability case. In this way parties are given the opportunity to form a judgment about whether they could hold another party liable for certain caused damage or if they could be held liable themselves. Also, parties can determine how much room for negotiation is available when settling the damage. In particular this information is important, since disputes are most frequently settled between parties themselves, rather than in court or with support of a (legal) expert(s) [12]. In addition, this information might be used to evaluate the legal advice of the attorney or an other legal professional. Naturally, it also helps parties to decide whether it is beneficial to take their case to court. Ideally, at the beginning of the negotiations, parties have an idea of what the outcome would be if their dispute would be decided by a judge. By information about previous court decisions, where relevant taking into consideration other factors such as time, costs, emotions, etc.,

a well-rounded impression is obtained about a parties' BATNA (Best Alternative To a Negotiated Agreement), that is: the result that should ideally at least be reached in the negotiations (the threshold). The BEST-project aims to provide disputing parties in a stage before they seek professional assistance with information about their position in the negotiations, to assist them in the dispute, get information about the legal possibilities to claim compensations, etcetera. The target user group of the program will generally consist of laymen in the field of law, who want to get an insight into the legal aspects of their dispute. In this paper, we focus on the problem of bridging the gap between the description of a case as it might be produced by a layman and the legal terminology that is used in legal documents. In particular, we present an approach for automatically determining relevant law articles based on an abstract description of a legal case. The paper is organized as follows. In section 2 we review the aim and the scope of the BEST-project. In section 3 we introduce a formal ontology of Dutch tort law that has been developed in the project. The use of this ontology for determining relevant articles based on logical reasoning is presented in section 4. We conclude with a discussion of the work in section 5.

## 2. The BEST-Project

A concept central to the BEST-project is a BATNA [6][13]. Fisher and Ury introduced principled negotiation, which advocates separating the problem from the people. Fundamental to the concept of principled negotiation is the notion of Know your best alternative to a negotiated agreement (BATNA). The reason you negotiate with someone is to produce better results than would otherwise occur. Or, as [1] puts it:

> Settlements are truly informed and voluntary only if the parties choose them with a full understanding of their alternatives.

If you are unaware of what results you could obtain if the negotiations are unsuccessful, you run the risk of:

1. Entering into an agreement that you would be better off rejecting; or
2. Rejecting an agreement you would be better off entering into.

In their three step model Lodder and Zeleznikow set forth three basic stages for the effective resolution of online disputes [9]:

1. Determining a BATNA, which helps the disputing parties determine what will happen if the dispute is not resolved;
2. Allowing parties to communicate among themselves using dialogue techniques; and
3. Using game theory techniques that employ compensation/trade-off strategies to attempt to resolve remaining issues in dispute.

The BEST-project aims at facilitating the establishment of a BATNA (step 1), through intelligent disclosure of case-law. We do this by employing ontologies and ontology-based search and navigation, as has been developed in [10]. The legal domain we are looking at is damages disputes. Two questions are relevant here. First, damages are compensated only if the other party can be held liable. Whether this is the case depends on a number of factors such as the probability of the occurred event and the na-

ture of the damages. Once a (legal) person can be held liable, the second question is what compensation is reasonable. Both answers are relevant for determining a BATNA in damages disputes.

### *2.1. Aim and Approach*

The BEST approach to supporting BATNA establishment is based on a number of principles that distinguishes it from other existing approaches to IT support for legal decision making. The most basic principle is that the project will provide supporting technology that prepares a legal judgement instead of trying to come up with such a judgement by itself. This preparation will consist of

- supporting the user to describe a specific legal situation
- retrieving and ranking descriptions of court decisions on similar cases

These functions will be implemented using a combination of statistical text retrieval methods and knowledge-based techniques. In particular, the idea is to ease the use of documents retrieval systems. This will be achieved by providing technological solutions for the two aspects mentioned above, in particular:

- An ontology-based interface for creating and classifying case descriptions
- An analysis component for generating search terms based on the classification of the case description

These components will be implemented and a prototype will be implemented that uses a thesaurus-based document retrieval system using a test data set. Further, the use of the components for enhancing existing search solutions for legal documents will be investigated.

The basic idea of the BEST approach is to de-couple the task of creating a meaningful and complete description of the case at hand from the task of retrieving similar cases. The rationale for this choice lies in the nature of the different terminologies used by laymen and by legal experts. An ontology suited to provide the terminology for supporting laymen in describing cases significantly differs from an ontology suited for providing the basis for annotating legal documents. This difference not only lies in the different terminology used by laymen and experts but also in the required representations. While an ontology for creating structured case descriptions needs to provide the basis for describing complex configurations of situations, the ontology for annotating legal documents will focus on the use of different words for describing the same legal concept or situation. It is easy to see that these tasks require conceptually different representations.

Besides the technical issues raised above a de-coupling of the case description and the document retrieval has several conceptual advantages:

- Depending on the user group, there can be different ways of describing cases that require different ontologies as a source of basic terms.
- Depending on the available data sources, there can be different retrieval engines that require different knowledge structures to determine relevant documents.
- The system will profit from using existing thesauri and annotations and provide added value to these systems by enriching them with query formulation support in terms of case descriptions.

- The system is able to point out potential liable parties the layman user might not have been aware of.

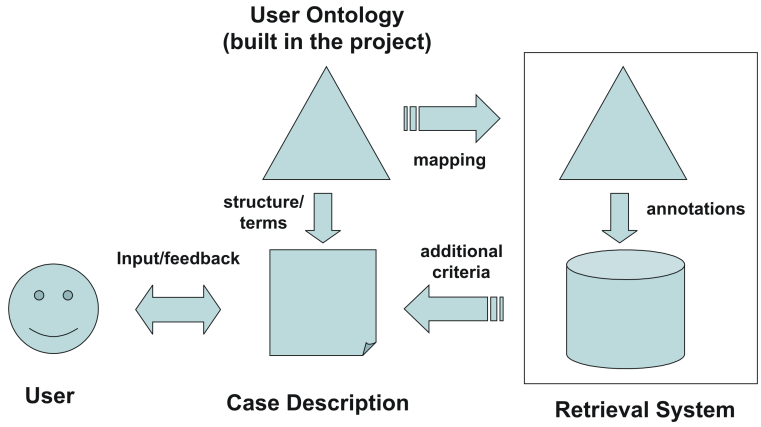


**Figure 1.** General Architecture of the BEST system.

Figure 1 shows a conceptual architecture that reflects the principle of de-coupling case description and document retrieval that will be the basis for the work in the BEST-project. The architecture is centered on the notion of a case description. The case description explicates relevant aspects of the case at hand using a structure and terms provided by a user ontology. This ontology is mapped on a second conceptual structure that is used to annotate legal documents.

### *2.2. Domain Scope*

The domain we will focus upon, is that of tort law. Whereas in common law systems various torts are distinguished (e.g., nuisance, fraud, negligence, trespass to chattels), civil law systems basically know one general tort action. In the Dutch Civil Code Article 6:162 reads, as translated by Betlem [4, p 291]

> Art. 6:162 BW. 1. A person who commits an unlawful act toward another which can be imputed to him, must repair the damage which the other person suffers as a consequence thereof. 2. Except where there is a ground of justification, the following acts are deemed to be unlawful: the violation of a right, an act or omission violating a statutory duty or a rule of unwritten law pertaining to proper social conduct.

Article 162.2 clearly illustrates the general nature of the unlawful acts that qualify as torts. Dutch law defines special cases (such as liability for children, employers, objects), but most of these refer (implicitly through the notion 'fault') to this general article 162.

Basically, four types of tort typically occur more often than others, which is interesting for us since the more cases, the better the technology used in the BEST-project works, are:

- Liability in traffic (verkeersaansprakelijkheid, art. 185 WVW)
- Liability of the employer for employees who get harmed in their working place (werkgeversaansprakelijkheid voor schade door de werknemer geleden, art. 7:658 BW)

The other two fall under the general Article 162:

- Medical faults (medische fouten, art. 6:162 BW)
- Offences of violence (geweldsdelicten, art. 6:162 BW)

A reason tort law is in principle suited for the BEST-project is that compared to other areas, the rights that determine the positions of both parties can mainly be found in the Civil Code and the case law based upon it. In contract law, for instance, parties often use special contractual clauses that are not similar to clauses that were covered in previous case law. In tort law the facts may differ, but the general legal concepts applicable to facts are largely similar, and can be found in statutes and case law. Both legal sources are electronically available.

### *2.3. Data Sources - Changing Insights*

The case law database used to disclose similar cases, is that of the public website www.rechtspraak.nl. For processing purposes we have all available 55.000 cases locally stored. Given the over 1 million legal verdicts annually, this is a low number. Nonetheless, this database contains almost all digitally available newer case law in the Netherlands. Even if commercial publishers would allow us access to their case law databases, only an additional 100.000 cases would become available electronically, most from before 1999 when the site Rechtspraak.nl was launched. We had hoped for higher numbers, and expected in particular to be able to work with a large number of cases from the lower courts. These cases are mostly not very interesting from a legal perspective, but for the BEST-project really useful. However, even the local courts do not have many cases available electronically. The relatively low number of cases available is one of the reasons that we believe it will be hard, if at all possible, to determine the amount of damages through case law retrieval. This would be different if we had as many cases as for example in the comparable Spanish project IURISERVICE [5], where in two years time over 2 million cases were collected. A possible way to partly meet this objection would be to use commercial case law databases besides the one of www.rechtspraak.nl.

Another problem is that in case law there often is no specific amount of damage compensation mentioned. The exact amount of damage usually is being determined in a separate procedure, the so-called 'schadestaatprocedure'.

All in all, to figure out the approximate amount of damage compensation that would be awarded in a court procedure will be probably done in a different way than initially thought. One option we consider is to use blind-bidding tools, such as Cybersettle.com and the like. These tools are quite popular in the insurance industry to settle cases where the only issue in dispute is the amount of damages. Beside this generic solution, for at least for one particular damage type another solution to this end is possible. With respect to injury compensation the damage compensations awarded in cases of an injury are collected in the Netherlands in the so-called 'Smartengeldgids'. We could include the information contained in this damages guide in our program to be developed. At this moment we already built an ontology based on this 'Smartengeldgids'. Damages concerned are divided into two subclasses: damage caused by an injury (injury to body or psychological damage) and damage not caused by an injury (privacy, false imprisonment etc.). All possible injuries of the first subclass are described. The domain of damages concerned with people alone consists of 189 classes.

## 3. An Ontology for Case Descriptions

There are three main domains integrated in the user ontology.

*Tort law* (hierarchy of article numbers and grounds for liability); When modeling the domain of tort law into the ontology, we mainly copied to the ontology the structure of tort law as it is described in Dutch law. The reason for this is that this structure already implies some consequences with regards to legal aspects. For instance, in Dutch tort law, there is a distinction made between direct liability and indirect liability. Further examples of liability are liability for actions of a child or possession of an animal and so forth.

Direct liability covers cases in which a person himself commits an unlawful act against another person. This unlawful act has been committed due to this persons' own fault. This is not the case when a person is liable for cases covered by indirect liability. Indirect liability is divided into liability for persons (like children or employees) and liability for objects (like animals or faulty products). When someone has a child less than fourteen years old, this person can be held liable for all damage that the child causes, whether the parent could do something about it or not.

This means that when someone describes a certain case in a user interface, the user ontology should be able to 'recognize' the various legally relevant aspects of this case. When a child of 15 causes damage, under Dutch law it is harder to hold its parents liable than when this child would have been 12 (in figure 2 we can see the different applicable article numbers).

*Entities subject to law* (juristic person, natural person, etc.); The division we made between natural persons and juristic persons was also made on grounds concerning legal implications. In the part where we discussed tort law, we already saw that it makes a difference if someone is a child or an adult. In the same way, it matters if an entity subject to law is a juristic person or a natural person (for example, a juristic person can never be an employee). Furthermore, in Dutch case-law it is decided that a public organization can never be regarded as a company. This means that the possibilities to hold a public organization liable are more limited than to hold a private organization liable. An overview of the division between natural persons and juristic persons, grounds for liability and the corresponding article numbers is given in figure 2.

*Objects in tort law* (motor vehicles, animals, product etcetera). For several objects, Dutch tort law provides a specific article concerning liability for damage caused with that object. In these different articles, different conditions apply regarding the liability. Therefore, it is important to know whether the object that caused the damage is an animal, a motor vehicle etcetera.

The ontology is completely modeled in OWL and currently contains about 300 classes most of which are actually defined in terms of logical axioms and 50 relations covering the most important aspects of the area of law introduced above as well as common sense terms for describing cases. More information about the ontology can be found in [11]. The ontology is available at http://www.best-project.nl/ontology/.


**Figure 2.** Partial class hierarchy of the user ontology.

## 4. Reasoning about Liability

A case can be described by defining individuals in the ontology: these are instances of concepts/classes and their relations or properties. For example, a horse is an instance of the class animals. The horse has the property causes with instance broken leg. Where broken leg is an instance of the class physical damage. Broken leg has the property affects with instance woman. Woman is an instance of the class plaintiff. Riding school is an instance of the class legal person. Riding school has the property owns with instance horse. The corresponding complete case description can be seen in figure 3.

The idea of the BEST approach is now to use the ontology to automatically determine the potential liability of actors in a case description. For this purpose, we extended the ontology about tort law with logical definitions of the different classes representing liability. In the following we briefly introduce the logical language used and give examples of how the logic was used to support automatic determination of potential liability. For a previous example of this approach see [7].

### *4.1. Description Logics*

Description Logics [2] are a special type of logic that is tailored to define terminological knowledge in terms of sets of objects with common properties. Recently, description logics have become popular as a formal foundation for the Web Ontology Language

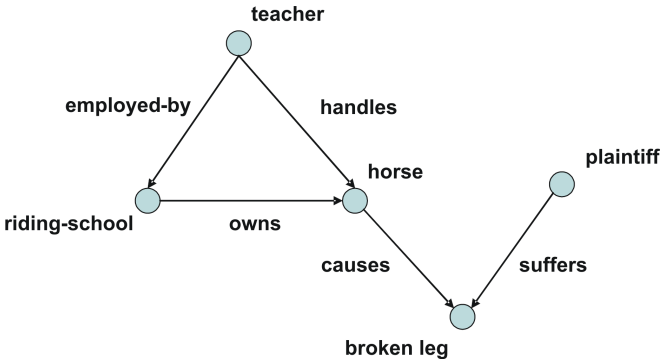


**Figure 3.** Case description as labeled graph.

OWL. The Basic modeling elements of a description logic are instances, concepts and relations. These modeling elements are provided with a formal semantics in terms of an abstract domain interpretation mapping $\mathcal{I}$ mapping each instance onto an element of an abstract domain $\Delta$. Instances can be connected by binary relations defined as subsets of $\Delta \times \Delta$. Concepts are interpreted as a subset of the abstract domain $\Delta$. Intuitively, a concept is a set of instances that share certain properties. These properties are defined in terms of concept expressions. Typical operators are the Boolean operators as well as universal and existential quantification over relations to instances in other concepts. The formal definitions can be found in the table below.

| **DL Expression** | **Semantics** |
|---|---|
| $A$ | $A^{\mathcal{I}} \subseteq \Delta$ |
| $\neg C$ | $(\neg C)^{\mathcal{I}} = \Delta - C^{\mathcal{I}}$ |
| $C \sqcap D$ | $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$ |
| $C \sqcup D$ | $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$ |
| $\exists R.C$ | $(\exists R.C)^{\mathcal{I}} = \{x \mid \exists y : (x,y) \in R^{\mathcal{I}}\}$ |
| $\forall R.C$ | $(\forall R.C)^{\mathcal{I}} = \{x \mid (x,y) \in R \implies y \in C^{\mathcal{I}}\}$ |

A Description Logic Knowledge base consists of a set of axioms about Instances, concepts (potentially defined in terms of complex concept expressions and relations). The first type of axioms can be used to describe instances. In particular, axioms can be used to state that an instance belongs to a concept, that two instances are in a certain relation. It is easy to see, that these axioms can be used to capture case descriptions as labeled graphs. The other type of axioms describe relations between concepts and instances. It can be stated that one concept is a subconcept of the other (all its instances are also instances of this other concept). Further, we can define a relation to be a subrelation or the inverse of another relation. These Axioms are used to formalize the legal ontology described in the last section. The formal definition of axioms can be found in the table below.

| DL Axiom | Semantics |
|---|---|
| $C(x)$ | $x^{\mathcal{I}} \in C^{\mathcal{I}}$ |
| $P(x,y)$ | $(x^{\mathcal{I}}, y^{\mathcal{I}}) \in P^{\mathcal{I}}$ |
| $C \sqsubseteq D$ | $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ |
| $P \sqsubseteq R$ | $P^{\mathcal{I}} \subseteq R^{\mathcal{I}}$ |
| $P \equiv R^{-}$ | $P^{\mathcal{I}} = \{(x,y) \mid (y,x) \in R^{\mathcal{I}}\}$ |

The formal semantics of concepts and relations as defined by the interpretation into $\Delta$ can be used to automatically infer new axioms from existing definitions. In particular, given an ontology and a number of instance related axioms, we can automatically determine whether an instance belongs to a certain concept based on the expression defining the concept.

### *4.2. Determining Potential Liability*

The idea of the automatic determination of potential liability is now to describe concepts related to liability in terms of necessary and sufficient conditions for liability using the logic introduced above. These conditions are phrased in terms of relations to other actors in a case. An example for such a definition is the following:

$$LiableForProperty \equiv \exists owns.(\exists causes.(damage \sqcap \exists affects.plaintiff))$$

It says that a person can in principle be held liable for property according to Dutch law if the person owns something that caused a damage to the accusing party. Using additional background knowledge from the different ontologies, this abstract definition can be matched with the case description shown in figure 3. The riding school owns the horse which caused an injury. Further, the accusing party suffered the injury. Using the additional information that an injury is a special case of a damage ($injury \sqsubseteq damage$) and that suffers is the inverse relation to affects ($suffers \equiv affects^{-}$), we can see that the riding-schoolfulfills all the necessary conditions of being potentially liable for property.

A characteristic of reasoning about potential liability is that often there are different parties that could be held liable and even the same party can often be held liable on different grounds. In the following, we show how these features are implemented in our approach using the concept of liability for a subordinate. This concept is defined in the following way:

$$LiableForSubordinate \equiv \exists hasSubordinate.PersonallyLiable$$

The definition says that someone can be held liable for a subordinate if there is a hasSubordinate relation to someone who is potentially personally liable in the case. Using the additional information that being employed is a special case of being a subordinate ($employs \sqsubseteq hasSubordinate$) we know that the riding school would also fulfill this requirement provided that the teacher, who is employed at the school is potentially personally liable in the case. This is checked by the system on the basis of the corresponding

definition of the concept of being personally liable. This concept is a quite generic one and covers a number of cases, only two of which we show here:

$$PersonallyLiable \equiv legal - person \sqcap \exists causes.(damage \sqcap \exists affects.plaintiff) \sqcup \exists handles.(object \sqcap \exists causes.damage \sqcap \exists affects.plaintiff) \sqcup ...$$

The definition says that someone is potentially personally liable if either he is a person in the legal sense and he has caused some damage that affected the accusing party or handles an object (in the legal sense) that has caused some damage to the accusing party, and so on. These two cases are interesting, because they require to explicitly make a distinction between legal persons and legal objects. If this distinction would not be made, the horse in our example would be classified as being personally liable. The definition above correctly determines that the teacher in our example is potentially personally liable according to the second case mentioned in the definition. Further, using the additional information that objects are not legal persons ($object \sqsubseteq \neg legal - person$) the system concludes that the horse is not liable for anything.

### *4.3. Our Axiomatization of Dutch Tort Law*

We performed a complete axiomatization of different forms of liability in Dutch tort law that allows us to automatically determine liability in complex case definitions. We tested the reasoning on a number of real world cases of which the one described above is the most simple using the FaCT reasoning system [8] in combination with the OilEd Ontology editor [3]. Some cases require very complex reasoning including relation hierarchies, inverse relations and global constraints and can take up to a couple of minutes. We are planning to move to a different reasoning system to improve the performance.

We tested the reasoning capabilities of the ontology on a number of example cases taken from legal text books. More details about these cases can be found in [11] and on the BEST web site (http://www.best-project.nl/cases/). The definitions of different forms of liability in the ontology was rich enough to cover almost all aspects of these example cases. We will discuss some of the current limitations in the next section.

## 5. Conclusions

Our claim is that logical reasoning can be used to mediate between the terminology used by laymen to describe legal cases and the terminology used in legal documents. The results presented in this paper provide a first important step in this direction. By being able to automatically detect relevant law articles based on a structured case description, we have made a first step towards determining relevant legal documents. The second step required is to link legal documents to combinations of law articles either by analyzing direct references to articles or through the occurrence of terms characteristic for a certain article.

While we have been able to build a computational ontology of a Dutch tort law that can determine relevant articles, the conclusions drawn based on this ontology are often impartial in the sense that certain relevant aspects of the case, such as knowledge

of dangers, attempts to minimize the danger, whether an object is faulty or not have been abstracted away in order to make the approach feasible. In order to really come up with a judgment on the case, these aspects have to be further investigated by a legal professional. We nevertheless belief that the judgement that can be done on the basis of the ontology is useful for the layman to better understand the options available. On the other hand, we avoid the danger of producing results that are pure speculation. A real evaluation of the approach will only be possible, however, when the second step – the retrieval of documents – is implemented.

## References


[1] M. C. Aaron. Finding settlement with numbers maps, and trees. In M. L. Moffit and R. C. Bordone, editors, *The Handbook of Dispute Resolution*. Jossey-Bass, 2005.

[2] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[3] S. Bechofer, I. Horrocks, C. Goble, and R. Stevens. OilEd: A reason-able ontology editor for the semantic web. In F. Baader, G. Brewka, and T. Eiter, editors, *KI 2001: Advances in Artificial Intelligence*, pages 396–408. Springer, 2001.

[4] G. Betlem. *Civil Liability for Transfrontier Pollution*. Graham and Trotman, 1993.

[5] P. Casanovas, N. Casellas, J.-J. V. M. Poblet, F. Ramos, J. Gorroogoitia, J. Contreras, M. Blï£¡quez, and R. Benjamins. Iuriservice ii: Ontology development and architectural design. In *Proc. of the Tenth International Conference on Artificial Intelligence and Law*, pages 188–194, Bologna, Italy, 2005.

[6] R. Fisher and W. Ury. *Getting to YES: Negotiating Agreement Without Giving In*. Haughton Mifflin, 1981.

[7] A. Gangemi, D. Pisanelli, and G. Steve. An ontological framework to represent norm dynamics. In R. Winkels, T. van Engers, and T. Bench-Capon, editors, *Proceedings of the Second International Workshop on Legal Ontologies*, 2001.

[8] I. Horrocks. Fact and ifact. In *Proceedings of the International Workshop on Description Logics (DL'99)*, pages 133–135, 1999.

[9] A. Lodder and J. Zeleznikow. Developing an online dispute resolution environment: Dialogue tools and negotiation support systems in a three-step model. *Harvard Negotiation Law Review*, 10:287–338, 2005.

[10] H. Stuckenschmidt, A. de Waard, R. Bhogal, C. Fluit, A. Kampman, J. van Buel, E. van Mulligen, J. Broekstra, I. Crowlesmith, F. van Harmelen, and T. Scerri. Exploring large document repositories with rdf technology - the DOPE project. *IEEE Intelligent Systems*, 2004.

[11] R. van Laarschot. Ontology-based knowledge modelling in dutch civil law. Master's thesis, Vrije Universiteit Amsterdam, August 2005.

[12] W. van Steenbergen. Doorverwijzing naar mediation vanuit de rechtspraak: Contradictio in terminis of een goede manier om de rechterlijke macht te ontlasten? *Nederlandsch Juristenblad*, 2005.

[13] J. Zeleznikow and A. R. Lodder, editors. *Fundamentals of Providing Negotiation Support Online: The Need for Developing BATNAs*. Wolf Legal Publishers, 2005.