Ontologies in AI and Law

Trevor Bench-Capon
Department of Computer Science
The University of Liverpool
Liverpool
UK

1. Introduction

In recent years the notion of ontologies has become increasingly important in the design and development of knowledge based and expert systems. This is certainly true of systems targeted at the legal domain, and a number of ontologies specifically for the legal domain have been developed. In this paper I will give an overview of ontologies and how they have been used in AI and Law.

I shall begin with a discussion of ontologies in general: what they are and why they are considered desirable. Next I shall discuss the various types of ontology. I shall then look at ontologies specific to AI and Law, both some of the early developments and some current approaches. I shall conclude with a discussion of some issues for ontologies and AI and Law.

2. Ontologies

The standard definition of "ontology" as used in Computer Science is that taken from Gruber (1993) "an explicit specification of the conceptualisation of the domain". There are several key points here:

- The ontology makes things explicit: without an ontology many design assumptions may be implicit in the executable representation.
- The ontology is supposed to formal: the notions it captures are thus precise and unambiguous.
- The ontology concerns some specific domain. This is perhaps debatable in that some ontologies attempt to represent common sense knowledge. The usual case, however, is that the ontology, like the system it supports, is directed towards a domain of application.
- The ontology represents a conceptualisation. Different people will conceptualise a domain differently according to experience, temperament and their tasks in the domain. We should therefore not expect that there is a single ontology applicable to a domain: the point of the ontology is often to make differences in conceptualisations visible.
- Related to this last point is that the ontology is very often directed towards some particular task in the domain.

From this general notion of an ontology we may ask what specifically we can expect as the components of an ontology. Ontologies typically contain representations and descriptions of:

- the types of *objects* found in the domain;

- the *attributes* which these objects may have;
- the *relationships* which these objects may enter into;
- *values* that the attributes may have for particular types;
- *axioms* constraining the above.

Not all ontologies will contain all of the above: all will represent the objects of the domain, but some ontologies will be more detailed than others.

There are a number of formal definitions of ontologies. One example, taken from Bench-Capon et al (2003) is:

Definition 1:

- Assuming a set of data types D
- An *ontology signature* is a pair <C,A>
  - – Where C is a partial order of *classes* (class hierarchy)
  - – A is a family of sets $A_{c,e}$ of *attribute symbols* where e is the type of the value of attribute a in c ($e \in C \cup D$)
- An *ontology* consists of a *signature* and a set of *axioms* which constrain the values of attributes.

For an example of a concrete ontology, consider the ontology for a motor car domain, shown in Figure 1, also taken from Bench-Capon et al (2003). Data types are in italics.

Model:
    Name: *String*
    Type: *{estate, saloon}*
    Hatchback: *{y,n,n/a}*
    Picture: *Gif*
    Rear-space: *Metres*^{2}
    Manufacturer: Motor-maker
    Price: *Pounds*

Car:
    Colour: *{white,red,blue}*

Motor-Maker
    Name: *{maker1, maker2}*
    Dealer: Dealer

Dealer
    Name: *String*
    Address: *String*

Axiom: hatchback(M) = n/a if type(M) = saloon

Figure 1: Example Ontology

From this ontology we can see, for example, that colour is a property of individual cars rather than models, and that only three colours are recognised. In a different conceptualisation, where the focus was on colour, we might find a longer list of precise shades, such as pillar box red and ocean blue. The key purpose of the ontology is to provide a vocabulary with which to model the information and knowledge to be put in the system, and to frame the input to and output from the system. The ontology determines what can and what cannot be said in an application: thus from the above example we can ask about the address of a dealer for a particular model, but we cannot ask anything about the speed of a car. The role it plays for a knowledge base is akin to that played by the conceptual schema of a database.

3. Uses of Ontologies

A number of roles have been suggested for ontologies. All of these are desirable, but different ontologies will place different emphases on them, and this may well have implications for the design of the ontology. In this section I will review the five main roles that have been identified as opportunities for ontology support.

3.1 Knowledge Acquisition

Knowledge acquisition is both essential to constructing a knowledge based system – which is only as good as the knowledge it contains – and rather difficult. The knowledge acquisition bottleneck has long been recognised as a major stumbling block to the widespread deployment of knowledge based systems. In order to carry out the task it must be possible to proceed in a systematic way, and to put the knowledge acquired into a coherent structure.

An ontology – which can be considered as the specification for a knowledge base – provides a very helpful framework to drive the knowledge acquisition process.

- It tells us precisely what it is we need to acquire knowledge about, and what knowledge we need to acquire about these things, and what information we can afford to ignore.
- The structure of the ontology provides a structure for the acquisition process.
- The ontology will identify gaps in the acquired knowledge that need to be completed, and help to determine when the acquisition process is complete.
- The ontology can be used to detect ambiguities and inconsistencies. This is particularly useful when we have multiple experts who may have subtly difference perspectives on the domain.

A leading example of an ontology targeted at knowledge acquisition, which has been successfully used in a number of applications, is Protégé (e.g. Noy et al 2000).

3.2 Knowledge Sharing

The second motivation for ontologies is to provide the ability to share knowledge between applications. Since it is often the case that several applications will have knowledge requirements in common, and because knowledge acquisition is not easy, it does seem attractive to have the ability to make use of knowledge already represented in another application. The difficulty, however, is that there is no guarantee that the knowledge will have been represented in the same way: for example, one knowledge base may have a three place relation parent, and indicate father and mother by the position in the relation. Another knowledge base may have two two place parent relations for each person, and distinguish fathers from mothers by using a predicate giving gender. In order to share knowledge these mismatches must be reconciled. This in turn requires that we have the kind of description of the vocabularies of the applications that an ontology provides. Thus ontologies are expected to provide the means to harmonise the representations of the different knowledge bases. The role of ontologies here is like that of an integration schema in database. A current example of knowledge sharing is the semantic web, which focuses a lot of current thinking about ontologies.

3.3 Knowledge Re-use

Problems quite similar to those arising from knowledge sharing occur when we wish to reuse knowledge developed for one application in another. Here the problems can be simpler, in that we are not dynamically accessing knowledge embedded in another system, but rather adapting an existing knowledge base. None the less we do require that the design assumptions and the intended meaning of the terms used in the knowledge base are explicitly available. Again we can find the documentation we need in an ontology. An example of knowledge reuse if found in Visser (1995).

3.4. Verification and Validation

Much verification and validation of knowledge based system simply involves inspection of the rules by experts, or else expert opinion on the outputs. Principled validation and verification requires that we have some independent specification against which the system can be validated. It has been argued that ontologies can play this role by making objective the aims of, and the constraints on, the system. A discussion of the role of ontologies in validation and verification can be found in Bench-Capon and Jones (1999).

3.5 Domain Theory Development

Often knowledge based systems are built not so much with the aim of producing a significant application, but with gaining understanding of the reasoning processes of some domain. It is often, however, hard to comprehend the system from the implementation itself. Equally it is often hard to make intelligent comparisons of two implementations of different approaches. If, however, we have a description of such systems in terms of their ontologies, then understanding and comparison can be achieved at the level of the conceptualisations: now the differences between the approaches are quite explicit, and this facilitates an explanation of their strengths and weaknesses. In Case Based Reasoning in Law for example, this level is useful in comparing, for example the factors used in CATO (Aleven 1997) and the dimensions used in HYPO (Ashley 1990), or in comparing CATO with rule based reconstructions such as Prakken and Sartor (1998). Constructing an ontology to articulate the assumptions underlying an implemented system can thus give some insight into understanding the domain in which it operates.

4. Types of Ontology

Within the basic definition of an ontology as given above there is considerable scope for variation. This is quite reasonable: given the potential variation in motivations as discussed in the previous section, it is unsurprising that different styles of ontology should develop. In this section I will discuss some of the main variants.

4.1 Lightweight Ontologies

At its simplest, an ontology may consist simply of a set of hierarchically organised terms. Such an ontology will resemble a thesaurus as used in information retrieval systems for a considerable time. The purpose of such an ontology is mainly to assist in information retrieval, the hierarchy permitting terms to be broadened or narrowed according as to whether too few or too many hits are obtained from the original query. The best known and most widely used such ontology is Wordnet (Miller 1990).

4.2 Upper or Top Ontologies

An upper or top ontology attempts to describe fundamental categories applicable to all domains. It will thus act as the pinnacle of a hierarchy and is intended to be elaborated in order to make explicit the particular domain concepts falling within these fundamental categories. Typically an upper ontology will start with a category such as "thing" and then descend through categories such as tangible and intangible thing, but will stop short of specific things such as cars and hope. Events, individuals and relations, and perhaps concepts relating to time and action are also likely to be found in such an ontology. The best known such ontology is CYC (e.g. Guha et al 1990).

4.3 Core or Domain Ontologies

A core, or domain, ontology attempts to articulate the concepts fundamental to some particular domain. Thus in law we would expect such an ontology to contain things such as statute, legal person and norm. There have been several such ontologies developed in law: the functional ontology of Valente (1995) and the frame based ontology of van Kralingen (1995) are examples.

4.4 Application Ontologies

An application ontology contains the very detailed and specific concepts required to perform a particular task on a particular piece of law. Such an ontology will contain notions such as "period of interruption of employment", "employer", "employee" and the like. In the legal domain such ontologies are often termed *statute specific ontologies* since they represent the concepts and terms used in a specific statute. Visser (1995) provides an example of such an ontology developed for the domain of Dutch Unemployment Benefit.

It is possible to see an application ontology as the bottom of a hierarchy in which an upper ontology is developed into a domain ontology which is further developed into the application ontology. All such ontologies typically contain the detail of attributes, values and axioms not found in lightweight ontologies. The coverage of a lightweight ontology, however, typically embraces all of these three levels.

5. Ontologies in AI and Law

In this section I will focus specifically on ontologies developed for use within AI and Law. I will begin by discussing two precursors to ontologies, which share the motivations, but which pre-date the popularisation of the use of the word. I shall then look at some early attempts consciously to produce an ontology. Finally I shall discuss some current examples of ontologies in AI and Law.

5.1 Precursors

Perhaps the first attempt to provide an ontology for AI and Law was the Language for Legal Discourse (LLD) developed by Thorne McCarty (1989). McCarty's aim was to provide a language which would mirror the structure of legal language, and hence be suitable for representing legal knowledge. The language included, in addition to atomic formulae, a number of rule constructs, horn clauses, horn clauses with embedded implications, horn clauses with embedded negations, default rules, prototypes and deformations. Additionally, and crucially for McCarty's programme, it addressed the problem of representing modalities, particularly those relating to time, events, actions, and the deontic notions of permission and obligation. This language thus has a lot of what would be expected in a top ontology. With regard to legal concepts McCarty argued that definitions were not enough because legal concepts are incurably open textured, change over time, and are created through a process of theory construction. He therefore conceptualises legal concepts using a set of invariant necessary conditions, a set of exemplars of the concept and a set of methods to transform between the exemplars. For an indication of how LLD might be developed into a domain ontology, one can consider its use to represent the concept of ownership in McCarty (2002).

Another early attempt to supply a formalism to give an ontological basis to legal was Ronald Stamper's Norma, described in Stamper (1991). Stamper hoped that this formalism could be used to model social systems, of which law he saw as one. Norma has three important ontological concepts:

- agents: which modify and regulate the world through actions for which they are responsible;
- behavioural invariants: which are associated with and characterise agents; and
- realisations: which are situations brought about through the performance of actions.

Despite the sophistication of this approach it was not widely used in AI and Law.

5.2 Early Ontologies in AI and Law

By the middle of the nineties, explicit ontologies for the legal domain were being constructed. I will mention two here: the Functional Ontology of law produced by Andre Valente at the University of Amsterdam (Velente 1995) and the Frame Based Ontology of Robert van Kralingen and Pepijn Visser at the University of Leiden (vam Kralingen 1995, Visser 1995).

In Valente's ontology law is conceived as an instrument to change or modify behaviour to realise social goals. He distinguishes six categories of legal knowledge:

- *normative knowledge* which attributes normative status to situations;
- *world knowledge* which describes the world that is being regulated;
- *responsibility knowledge* which ascribes responsibility for the violation of norms;
- *reactive knowledge* which describes sanctions for the violation of norms;
- *meta-legal knowledge* which is used to resolve conflicts;
- *creative knowledge* which states how legal knowledge is created

These categories are intended to provide a top ontology with which to classify the various elements that are required to make up a functioning legal system. In most applications it is the transition between normative and world knowledge that attracts most attention: Valente was insistent that the other kinds of knowledge should not be neglected.

The frame based ontology developed at Leiden brought together the work of two researchers: van Kralingen developed a generic ontology and Visser used this to develop a statute specific ontology describing Dutch Unemployment Benefit Law. The generic ontology distinguished three kinds of entity: norms, actions and concepts. For each of these it identified the attributes associated with the entity, and was thus able to serve as a template for knowledge acquisition, intended to be applicable to any legal application domain. The statute specific ontology would then describe the particular norms associated with a statute, the actions which they regulated and the concepts involved in these actions and norms. Essentially this specified the vocabulary with which any particular application based on this statute could be represented. In Visser (1995) two applications are produced: determination of benefit and planning. The general system development methodology based on this ontology is described in van Kralingen at al (1999).

5.3 Current Ontologies in Law

Since the early work on ontologies in AI and Law, ontologies have increasingly formed part of systems development. In this section I will briefly mention four current examples to give a flavour of this work.

The *Norme in Rete* project (Bolioli et al 2002) is done for the Italian Research Council and the Italian Ministry of Justice. It is an example of a lightweight ontology which aims to define and promote a controlled language for legislation. It is intended to support drafting and version control of legislation, and by providing a standard for terminology should promote more effective information retrieval.

The E-Court ontology (Breuker et al 2002), developed at the University of Amsterdam supplies an upper ontology targeted at Dutch Criminal Law. This detailed ontology is especially designed to support the knowledge management of documents used in Dutch courts.

Perhaps the most complete and ambitious example of the use of ontologies in the development of legal information systems is the E-POWER project (e.g. van Engers et al 2001). This project, undertaken for the Dutch Income Tax organisation, supports all stages of legislative activity from drafting, through promulgation to executable systems to apply the legislation, using a connected set of ontological models. The ability to detect anomalies and inconsistencies during drafting is especially interesting. This is an exciting project of enormous potential.

Finally, as an example of a application specific ontology we can consider iPROnto (Delgado et al 2003). This ontology is intended to support software agents in the management of digital rights. In is based on SUMO, from the IEEE Upper ontology initiative, and draws its specific ontology from the World Intellectual Property Organisation framework for Intellectual Property Rights.

6. Discussion

In this section I will make some remarks on three topics. Firstly I will point to a shift in motivation that can be observed in the use of ontologies in recent years. I shall then discuss a central question in ontologies, *how task neutral can an ontology be?* Finally I shall conclude the paper with a few remarks on successful use of ontologies.

6.1 Motivation for Ontologies

The original motivation for ontologies was in connection with the development of knowledge based systems. The focus was thus on knowledge acquisition, and knowledge re-use. Ontologies produced here were typically upper ontologies, subsequently developed into application specific ontologies. The work of van Kralingen et al (1999) is representative of this approach. Currently, however, the availability of vast quantities of information through world wide web has switched the emphasis to knowledge sharing, and in particular support for the vision of the of the Semantic Web. In this context lightweight ontologies can be useful, although full scale ontologies will still be required if this vision is to be fully realised.

6.2 Do Ontologies Depend on Task?

This is an important question: the aspiration is to provide an ontology that will conceptualise a domain, and be applicable and useful in a range of applications within that domain. The problem, however, is that there is rarely a single conceptualisation of a domain, and the way in which a given person conceptualises the domain will depend on how that person came to understand the domain, and this is often a matter of particular experiences through the performance of particular tasks in the domain.

For example suppose we were trying to build two legal applications: one to deal with immigration regulation and one to deal with the regulation of currency movements. Suppose we take as out intended starting point the task neutral ontology of Wordnet. A fragment of its hierarchy describing countries is shown in Figure 1.

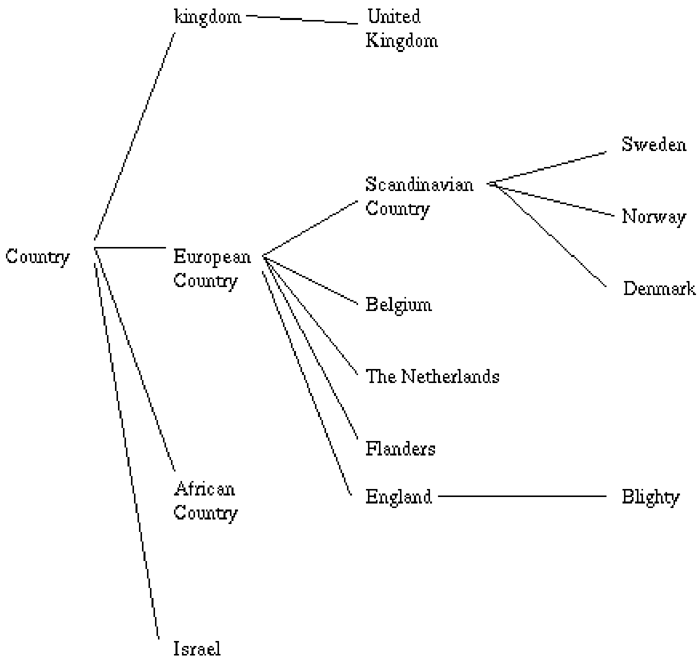

Figure 1: Ontology from Wordnet

Already we can see that there are some particular choices made here:

- "United Kingdom" is included as a hyponym of "kingdom" , while its constituent parts (England, Scotland, Wales and Northern Ireland) are included as hyponyms of European country. Thus UK is not a sibling of Italy, France and the other countries, which we probably need for our purposes, since it is the UK which controls immigration and currency, rather than its member countries.
- Is the division by continent appropriate? Those of us who regularly travel in Europe know that the broad division applied at immigration points is between European Union and non-European Union countries. (In passing, European Union does not appear in Wordnet, although some older terms such as "European Economic Community" and "Common Market" are given as synonyms and without hyponyms). On entering the US, on the other hand, the division is into the US and rest of the world.
- The divisions of European country are also interesting, but unhelpful. We find the following hyponyms: Scandinavian country, Balkan Country, and then individual European countries. Unfortunately this means that EU countries such as Sweden and France are not siblings, and non EU countries such as France and Switzerland are. Worse, no longer existing countries appear at this level: Flanders appears (described as "a medieval country in northern Europe") as a sibling of Belgium and the Netherlands. East Germany still appears as a republic in north central Europe. Other past countries such as Burgundy do not appear at all. Sometimes former names for countries are treated as synonyms (Ghana and Gold Coast), but sometimes as hyponyms (China, Cathay). One slang term for a country appears ("Blighty" as a hyponym of "England"), whereas others do not (no "Emerald Isle" for "Ireland". Fictional countries such as Ruritania (rightly?) do not appear.

An ontology for immigration would need to take account of treaties. The EU has differential rules for non-EU countries, and some countries have further agreements. Suppose one such agreement was the “Schvenigen” Treaty with the Netherlands, Belgium and Sweden as signatories. An ontology form immigration would look as in Figure 2.

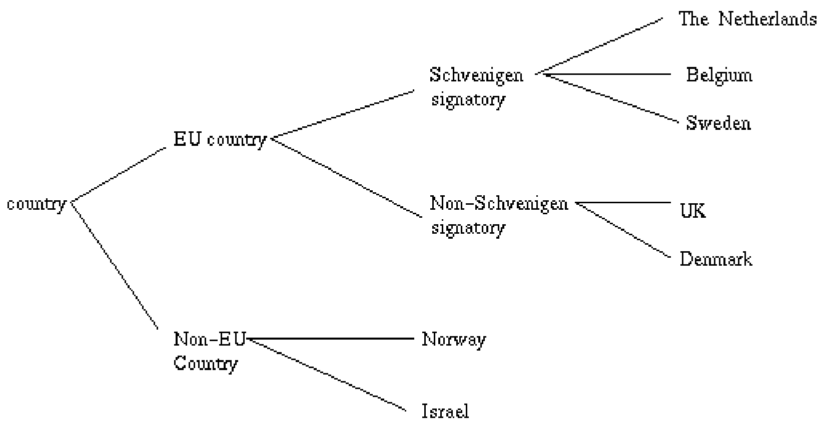

Figure 2: Ontology for Immigration

There are a number of differences here. Concepts not included in Wordnet are required; concepts, such as the component countries of the UK are not required; and the semantic distance between countries changes: for example, Norway is a sibling of Israel in Figure 2, and its great-nephew in Figure 1.

Similarly an ontology for currency regulation would need to know about different currency areas. (Supposing, for the sake of the example Sweden is a sterling country and Norway is a Euro country).

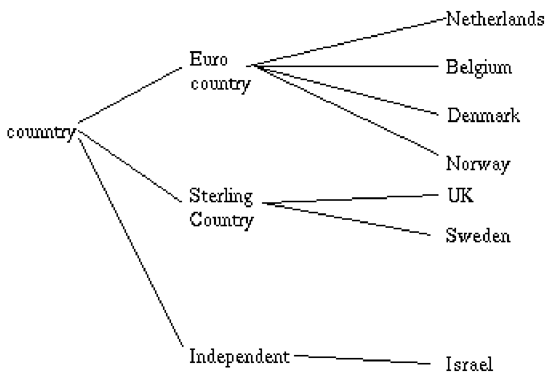

Figure 3: Ontology for Currency Control

Once more we have different concepts, and once more the semantic distance changes: Israel is now the cousin of Norway.

These examples serve, I believe, to show the great difficulty in coming up with a sensible ontology with reference to some specific task. In general, multi-task ontologies such as Wordnet tend to be rather shallow with high fan out, whereas for building particular applications, more depth with low fan out tends to be more useful. Second, the task determines the attributes in which we are interested: if we attempt to anticipate a range of applications we will introduce redundancy, and almost certainly fail to foresee some needs. I would argue that Wordnet may well be useful for information retrieval, but cannot be seen as the task neutral basis for a range of information systems.

6.3 Successes

I will conclude on a positive note by pointing to some successes of ontologies. First I would claim that they have proved their worth, particularly in information retrieval applications, but also in system development: the E-POWER project referred to above is a significant achievement. Second I would claim that something like an ontology is now seen as an essential component of any principled system development: one would no more build a knowledge based system without one, than one would develop a database without a data dictionary.

Finally ontologies are central to the vision of the semantic web. This hold out very exciting prospects and would be impossible without ontologies. The future will let us see whether it is possible with them.

References

Aleven, V. (1997). *Teaching Case Based Argumentation Through an Example and Models.* PhD Thesis. The University of Pittsburgh.
Ashley, K.D., (1990). *Modeling Legal Argument*. MIT Press, Cambridge, MA.

Bench-Capon, T., and Jones, D., (1999). *PRONTO: Ontology Based Evaluation of Knowledge Based Systems* in A. Vermesan and F. Coenen (eds) *Validation and Verification of Knowledge Based Systems*. Kluwer Academic Publishers: Dordrecht.

Bench-Capon, T.J.M., Malcolm, G., and Shave, M.J.R., (2003). Semantics for Interoperability: Relating Ontologies and Schemata. V. Marik, W. Retschitzegger, O. Stepánková (Eds.): Database and Expert Systems Applications, 14th International Conference, DEXA 2003, Lecture Notes in Computer Science 2736 Springer: 703-712.

Bolioli, A,, Dini, L., Mercatali, P., Romano, F., (2002), For the Automated Mark-Up of Italian Legislative Texts in XML. In T. Bench-Capon, A. Daskalopulu and R. Winkels (eds), *Legal Knowledge and Information Systems JURIX 2002: The Fifteenth Annual Conference*, IOS Press Amsterdam, p21-30.

Breuker, J., Elhag, A., Petkov, E., and Winkels, R., (2002). Ontologies for Legal Information Serving and Knowledge Management. In T. Bench-Capon, A. Daskalopulu and R. Winkels (eds), *Legal Knowledge and Information Systems JURIX 2002: The Fifteenth Annual Conference*, IOS Press Amsterdam, p73-82.

Delgado, J., Gallego, I., Llorente, S., and Garcia, R., (2003). iPROnto: An Ontology for Digital Rights Management. In D. Bourcier (ed) *Legal Knowledge and Information Systems JURIX 2003: The Sixteenth Annual Conference*, IOS Press Amsterdam, p111-21.

van Engers, T.M., Gerrits, R., Boekenoogen, M., Glassée, E., Kordelaar, P., (2001). POWER: using UML/OCL for modeling legislation - an application report. In *Proceedings of the 8th International Conference on Artificial intelligence and Law*, ACM Press: New York, p157-67.

Gruber, T.R., (1993). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993.

Guha, R. V., Lenat, D. B., Pittman, K. , Pratt, D. , and Shepherd M., (1990). Cyc: A Midterm Report. *Communications of the ACM* 33 , no. 8.

van Kralingen, R.A., (1995). *Frame Based Conceptual Models of Statute Law*. Kluwer Law International: The Hague.

van Kralingen, R., Visser, P.R.S., Bench-Capon, T.J.M. and van der Herik, J., (1999). *A Principled Methodology for the Development of Legal Knowledge Systems*, International Journal of Human Computer Studies, 51: 6 pp 1127-54.

McCarty, L.T., (1989). A language for legal Discourse I. basic features in *Proceedings of the second international conference on Artificial intelligence and law*, ACM Press: New York, pp180-9.

McCarty, L.T., (2002). Ownership: A Case Study in Representing Legal Concepts. *AI and Law*, volume 10, nos 1-3 pp 135-61.

Miller, George A., Richard Beckwith, Christiane Fellbaum, Derek Gross and Katherine J. Miller. 'Introduction to WordNet: an on-line lexical database'. In: *International Journal of Lexicography* 3 (4), 1990,

Noy, N.F., Fergerson, R.W. and Musen, M.A., (2000). The knowledge model of Protege-2000: Combining interoperability and flexibility. *12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000),* Juan-les-Pins, France, 2000.

Prakken, H., and G. Sartor 1998. Modelling Reasoning with Precedents in a Formal Dialogue Game. *Artificial Intelligence and Law* 6: 231-287.

Stamper, R.K., (1991). The role of semantics in legal expert systems and legal reasoning. *Ratio Juris* 4(2): pp219-44.

Valente, A., (1995). *Legal Knowledge Engineering: A Modelling Approach*. IOS Press: Amsterdam.

Visser, P.R.S., (1995). *Knowledge Specification for Multiple Tasks.* Kluwer Law International: The Hague.