Legal knowledge based systems
JURIX 94
The Relation with Legal Theory

The Foundation for Legal Knowledge Systems
Editors:
H. Prakken
A.J. Muntjewerff
A. Soeteman

N. den Haan and R. Winkels, The Deep Structure of Law, in: H. Prakken, A.J. Muntjewerff, A. Soeteman (eds.), Legal knowledge based systems JURIX 94: The Foundation for Legal Knowledge Systems, Lelystad: Koninklijke Vermande, 1994, pp. 43-54, ISBN 90 5458 190 5.

More information about the JURIX foundation and its activities can be obtained by contacting the JURIX secretariat:

Mr. C.N.J. de Vey Mestdagh
University of Groningen, Faculty of Law
Oude Kijk in 't Jatstraat 26
P.O. Box 716
9700 AS Groningen
Tel: +31 50 3635790/5433
Fax: +31 50 3635603
Email: sesam@rechten.rug.nl

The Deep Structure of Law

Nienke den Haan and Radboud Winkels
Dept. of Computer Science & Law, University of Amsterdam
Kloveniersburgwal 72, 1012 CZ Amsterdam, The Netherlands
E-Mail: {nienke,winkels}@lri.jur.uva.nl

Abstract

This article discusses theoretical research on abstraction methods that describe the contents of laws at an abstract level above the code. Abstractions are useful for several purposes, for instance to determine the normative essence of an existing law, which is interesting from the viewpoint of legal theory. In legislative drafting deep structures are intermediate formalizations between the normative intentions of the lawgivers and the possible ways to codify the intentions. A tentative framework is given for the construction of deep structures out of structural elements, and for the generation of different legislative paraphrases with identical normative effects. We can use deep structures in legislative drafting as a formalization and modelling step between the phase in which global normative requirements are formulated for behaviour which is to be regulated, and the phase in which codification takes place.
Keywords: Modelling, legislative drafting, extensional representation

1 Introduction

Both in legal theory (e.g. [vonWright, 1981; Brouwer, 1990]) and in the field of Artificial Intelligence and Law (e.g. [vanKralingen *et al.*, 1993; Hage, 1993]), people have stated that a legal norm as such, and its expression in rules are not the same thing and should be distinguished. Different regulations (statutory sources) may have the same effect when 'applied', i.e. a set of norms may have many paraphrases. We hypothesize a 'deep structure' of legislation that reflects the normative effects of different paraphrases.[1] Deep structures capture the semantics of a regulation, whereas surface structures (see Section 3) are syntactic expressions of the deep structure.

In this article we discuss the structure of deep structures, and how they are built by modelling normative effects of regulations. To be able to test our ideas we choose normative problems in clear and small worlds. We will formulate deep structures and generate several paraphrases of possible regulations.

2 Classifying behaviour

That legislation is assumed to control behaviour, or possible states of affairs in the 'world' is recognized in the literature (e.g. recently in [Peters & Vlemminx, 1991]). Not

[1] Not to be confused with the deep structures of McCarty as discussed in [Zeleznikow & Hunter, 1994] which refer to the fact patterns of legal decision making.

the whole world is covered by legislation, or the world as we see it, but part of the world and a particular, legal abstraction of that world. E.g. in the world of traffic from a legislative point of view, there exist only motor vehicles, pedestrians, parts of the road, etc., not 'Volvo's', racing bikes, or asphalt roads. In such an abstracted, legal world, not all 'physically possible' behaviour is desirable, not all possible situations are as they 'ought to be'. Laws and regulations can be seen as prescribing or permitting desirable behaviour or situations, and/or prohibiting undesirable behaviour or situations (see e.g. [Waaldijk, 1985]).[2] The net effect of such regulations, i.e. the restriction of all possible behaviour, can be achieved in many different ways.

We will call the world of physically possible behaviour W_p and the world of desirable behaviour W_d, with $W_d \subseteq W_p$.[3] The difference between these two worlds we call W_u, the set of undesirable situations.

The union of desirable and undesirable situations must be equal to the world of all physically possible behaviour: $W_d \cup W_u = W_p$. Reversely, we have $W_d \cap W_u = \emptyset$. Therefore, one way to 'describe' W_d is by explicitly prohibiting situations that are undesirable, i.e. W_u. The rule(s) that implement the prohibition, may be somewhat too general, and some exceptions to the rule are specified (permissions). In other areas, the rule may be too specific, and extra rules are specified to prohibit more situations.

We can also start with a rule that explicitly permits or prescribes the desired situations (W_d). One can think of domains where the default is "illegal" or undesirable. For instance in safety related domains the risks at stake may be so high that in principle all behaviour is undesirable except for some limited prescribed courses of action. In such a case the regulation prescribes those actions (using obligations) and all others can be considered illegal. E.g. [Hammond *et al.*, 1994] describe the domain of safety protocols for clinical tests of medicine like chemotherapy treatment of cancer. The risk of killing a patient by wrong treatment should be minimized as much as possible, and the protocols (and regulations about protocols) are prescribing the narrow paths that lead to desirable situations (W_d). One can find the same approach in 'formal' or procedural law, e.g. rules guiding courses of action for trials often take the form of prescriptions and obligations rather than prohibitions.

The net effect of these two approaches can be the same, both resulting pieces of legislation (normative systems) can describe the same set of desired situations (W_d) and prohibit the same set of undesirable situations (W_u). They achieve the same thing, have the same "deep structure".

In this paper we will describe the nature of this deep structure and ways of generating paraphrases of it in regulations and illustrate our approach by means of a simple example. The running example of this paper concerns the problems with the use of the computers of our department. We perform the roles of both the legislative drafter and the codifier. We only want students of the Department of Computer Science and Law to use our computers, and we do not want them to use the machines for hacking, playing games or

[2]Of course not all possible behaviour that is undesirable needs to be forbidden by regulations. Some may be made impossible by restricting the physical world, e.g. equipping lorries with speed restriction devices. Other undesirable behaviour may not occur in reality because nobody will benefit from it or even think of it, etc.

[3]In principle it is possible W_d is equal to W_p, i.e. all possible situations are desirable or permitted. In that case no legislation is necessary, a white raven in the Netherlands.

commercial use.

3 The Surface Structures of Regulations

Each regulation is some expression of a deep structure. On the surface of the regulation we do not perceive this deep structure, but only superficial structural elements. The rules in a regulation have a specific structure, the intra-rule structure. Furthermore, the regulations contain structures defined over rules, the inter-rule structure. When we look at more than one regulation at the same time, we can also see a inter-regulation structure.

1. **Structures in legal rules:** When we take a superficial look at regulations, the first building blocks we discern are the legal rules or articles. The description of legal rules consists of an 'object level' description, i.e. a description of the objects, agents and actions concerned, and of a normative component, which is usually represented in some form or derivate of deontic logic.

   In intra-rule structures we see that legal rules consist of a description of its applicability grounds, which are usually represented as the conditions of the rule, and of normative descriptions, which are usually represented as the conclusions of the rule. For instance, if we have the rule: "The computers of our department may be used by students enrolled in one of the Computer Science & Law classes", we could construct a representation such as: $\forall_x \exists_y CSL\_Computer(x) \wedge CSL\_Student(y) \rightarrow Permitted(use(y,x))$ (see Section 5.1).

2. **Structures in regulations: Exceptions.** Important inter-rule structures are exception structures. Often the law text contains a general rule which prescribes a norm for a large group of individuals, actions or objects, followed by a rule that excludes specific members or subsets of these groups. The exception can be made *directly* in the same article, or *indirectly* in another paragraph or even another regulation. Furthermore, the exception can be made *explicitly*, characterized by words like 'contrary to' and 'except', or *implicitly* where the exception has to be inferred. For example, in our domain the requirement: "Only students may use our computer equipment, but not for commercial means" could be codified as: "`art 1.a: Only students may use the computers of the CSL department`", and "art 1.b: Use of the CSL computers for commercial means is prohibited". As an indirect and implicit exception, we may state in another paragraph on system management: "`art 23: The system manager may grant permission to an individual to use the CSL computers for a specific goal`". Article 23 acts as an implicit exception to article 1.a (and even 1.b), because any individual, and not just students may be granted access to the CSL computers (apparently for any use). This exception has to be inferred at the domain level from the agent hierarchies of computer users (see Section 4). Conflicts between rules because of such exceptions are solved by 'secondary rules' ([Hart, 1961]; see e.g. [Sartor, 1992] for a discussion on normative conflicts). For instance, in our domain we could define a domain specific secondary rule that states that decisions by the system manager prevail over general rules.

We can locate exceptions by studying deontic contradictions (see [Soeteman, 1981] for an extensive discussion of deontic contradictions). Contradictions may arise between norm subjects (e.g. Ox and $O\neg x$), and between the normative sentence (e.g. $\neg Ox$ and Ox). Contradictions between norm subjects are easily spotted when, as in this case, we see x versus $\neg x$. On the other hand, we can also determine by means of interpretation on domain knowledge, a contradiction between x and y, where x and y are in some way contradictory or incompatible, e.g. *commercial_use* versus *educational_use* in our example domain (see Section 4).

3. **Structures in regulations: References.** Another, less frequently used form of inter-rule structure is expressed in references between rules. In regulations we can find references of several orders of magnitude: firstly, we can see references to another article from the same regulation, such as: "`A notice of objection, as defined in article 7 sub 2 must be motivated and filed...`". In this case, the definition of article 7 sub 2 has been used. However, articles can also be referenced in order to form additions or rejections of argumentation. For example, the additional: "`Besides the definitions from articles 5 and 6, natural persons or parties directly harmed by a decision, may also appeal...`". References prevent that we have to repeat definitions or norms that we use later on.

4. **Structures over regulations:** Not only can we study the law texts separately, but together they form a complex structure which determines the legality of our behaviour. Each regulation is a carefully composed unit. Generally, each statute represents a set of rules concerning one particular subject. The legal hierarchy distinguishes national statutes, regulations, and amendments. Also, new legislation may prevail over, or add to old legislation. Just as rules define normative descriptions over parts of all possible situations, regulations define normative descriptions albeit possibly to a larger subset of possible situations. We can find exception and reference structures between regulations.

4 The Deep Structure of Regulations

The first thing we can say about the deep structure of regulations is that it will have to distinguish the 'world' the regulations try to rule from the normative requirements about situations or behaviour in that world, i.e. domain knowledge versus regulation knowledge (cf. [Breuker & denHaan, 1991]). The same distinction was found at the surface level of regulations (section 3).

1. **A Model of the World to regulate:** *Terminological knowledge:* Objects/agents and actions with attributes (e.g. *is_a* relations).

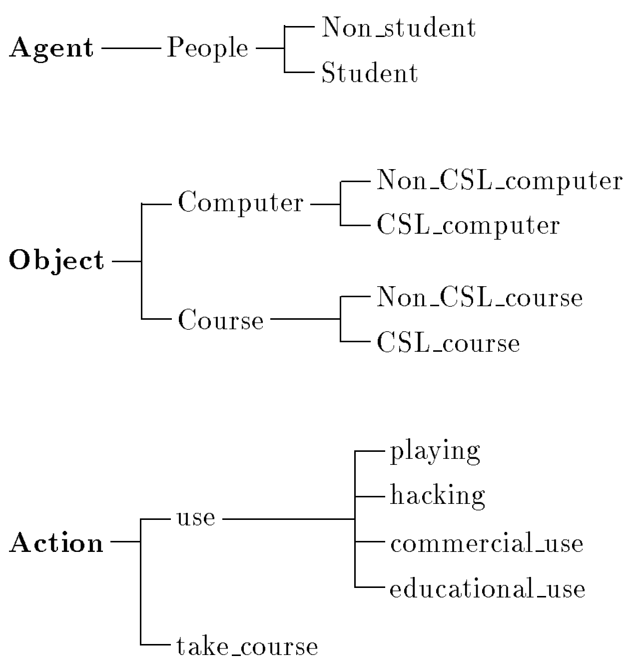
   Behavioural knowledge: Causal/intentional relations between objects and actions

   Structural knowledge: Structural relations between objects or actions (e.g. consist–of and spatial relations). In the traffic domain we see for instance: a road consists of pavement, driving lane, white line, driving lane, pavement. In our example domain we do not encounter structural knowledge.

2. Normative requirements to situations: desirable and undesirable. The normative definitions are expressed in legal rules, which generally prescribe norms to generic situations. A situation is a collection of behaviour descriptions in the world that can be seen as a single entity: an agent, object or an action. An example in the world of computer use would be the use of the terms $use(Person, Computer), take\_course(Person, Course)$ in a primary rule. Whereas we see specific descriptions of situations in cases, legal rules contain more general descriptions because they have to apply to classes of e.g. computers (as opposed to instances the instances such as *'SPARC_station_ELC'* we may see in a case[4]). The deep structure of legislation is an abstraction of the legal rules, and therefore contains generic descriptions and normative requirements. For each normative system there has to be a default normative requirement for a situation in the world. The default can be either 'legal' or 'illegal' (or 'desirable' versus 'undesirable'). In most cases the default will be 'legal', i.e. if in a particular regulation nothing is said about a particular situation, it is a legal situation. This distinction is expressed as: "`Everything is permitted but ...`" versus "`Everything is forbidden but ...`", i.e. we define the normative effect according to W_u or W_d.

For our example domain the deep structure could be constructed as follows:

Model of the world: *Terminological knowledge*[5]



Behavioural knowledge: As the world knowledge describes all possible behaviour on the world, and the normative definitions link norms to sets of behaviour, the world

[4]In applying the legislation to specific situations, the specific situation in a case is compared with the generic prescriptions in primary rules. For this paper we are not concerned with actually applying legislation.

[5]We are not concerned with attributes in this example.

knowledge is formalized using a set notation: $\{(co, st) \mid Course(co) \wedge Student(st) \wedge take\_course(co, st)\}$ denotes the set of all co and st where $take\_course(co, st)$ is the case, with co of type $Course$, and st of the type $Student$. The set of all possible generic situations is called W_p, and is defined over the possible types of behaviour as:

$$W_p = \{(c, p, co) \mid Computer(c) \wedge People(p) \wedge Course(co) \wedge use(c, p) \\ \wedge take\_course(co, st)\}$$

which reads as: the set of all behaviours use and $take\_course$ which are defined over all computers and people, and all courses and people respectively.

Structural knowledge: None

Normative requirements: In our example we only want students to use our computers for educational use. We can formalize this in two different ways, starting with W_d or starting with W_u. Reasoning from the desirable state, we allow situations where our own students use our computers for educational means. We do not pose any restrictions on the use of other computers than our own:

$$\begin{aligned} W_d = \{\{(co, st, c) \mid\ & CSL\_Course(co) \wedge \\ & Student(st) \wedge \\ & CSL\_Computer(c) \wedge \\ & take\_course(co, st) \wedge \\ & educational\_use(c, st)\} \\ \cup \\ \{(c, p, co) \mid\ & Non\_CSL\_Computer(c) \wedge \\ & People(p) \wedge \\ & Course(co) \wedge \\ & use(c, p) \wedge \\ & take\_course(co, p)\} \\ \} \end{aligned}$$

If we look at the restrictions on the computer use of students, we see that we want to forbid students to use our equipment for Non_educational purposes. Reasoning from the undesirable state, we therefore describe situations in which our students play, hack, or use our computers for commercial means. Secondly, we forbid anyone else to use our computers:

$$\begin{aligned} W_u = \{\{(co, st, c) \mid\ & CSL\_Course(co) \wedge \\ & Student(st) \wedge \\ & CSL\_Computer(c) \wedge \\ & take\_course(co, st) \wedge \\ & playing(c, st)\} \\ \cup \\ \{(co, st, c) \mid\ & CSL\_Course(co) \wedge \\ & Student(st) \wedge \\ & CSL\_Computer(c) \wedge \\ & take\_course(co, st) \wedge \\ & hacking(c, st)\} \end{aligned}$$

$$
\begin{aligned}
&\cup \\
&\{(co, st, c) \mid CSL\_Course(co) \wedge \\
&\qquad Student(st) \wedge \\
&\qquad CSL\_Computer(c) \wedge \\
&\qquad take\_course(co, st) \wedge \\
&\qquad commercial\_use(c, st)\} \\
&\cup \\
&\{(c, co, p) \mid CSL\_Computer(c) \wedge \\
&\qquad People(p) \wedge \\
&\qquad Course(co) \wedge \\
&\qquad take\_course(co, p) \wedge \\
&\qquad use(c, p)\} \\
\}
\end{aligned}
$$

If we look at the union of W_d and W_u, indeed we see that we have W_p, and $\emptyset$ for the intersection of W_d and W_u.

5 Generating Paraphrases

Given the deep structure of a normative system, we can start generating surface structure expressions that reflect the deep structure. For each generic situation for which normative qualifications are defined in the deep structure, legal rules are constructed. Below we present several dimensions and mechanisms for generating different surface structures. The resulting surface structures differ with respect to readability, maintenance, learnability, and the suitability for specific purpose or target group.

The first choice that has to be made is whether the *default* will be 'illegal' or 'legal', whether we will describe W_d or W_u. The normative part of the deep structure is already based on a choice, but in principle it is possible to generate a surface structure that starts from the opposite assumption. For some purposes it may even be advisable to describe both W_u and W_d, e.g. if the legislation should be understandable for small children that need to know both what is permitted and what is forbidden, even though one could infer one from the other. In most cases at least some rules will refer to situations that are actually covered by the default for ease of understanding.

Other choices will have strong implications for the structure of the resulting set of rules. We have to choose a *viewpoint* for describing normative positions of situations. The different possible viewpoints correspond to the different types of knowledge in the world model of the deep structure. The viewpoint can be based on the agents and objects in the world, on the possible behaviour, and on structures in the world. For our example domain, we could describe the deep structure from the perspective of using computers (i.e. what use is desirable, what use is not), or from the perspective of users (i.e. which users can use the computers, which can't). The results of different choices can be dramatically different on the surface. One viewpoint may result in mostly explicit exceptions, another in mostly implicit exceptions for instance. This again has consequences for the readability and learnability of the legislation. Explicit exception structures are easier to read and learn for humans than implicit ones.

A third dimension for generating different expressions of the same deep structure is the level of *abstraction* one chooses for the surface structure. When we have chosen a particular level of detail, we can only make additions on lower levels which can be abstracted to the chosen level. E.g. if we have two instances a and b which can be abstracted to c, and the normative effects Fx and Fy which can be abstracted to Fz, we can make the following abstractions :

$\{a \rightarrow Fx, b \rightarrow Fx\} \stackrel{abs.}{\rightarrow} a \vee b \rightarrow Fx$ or:

$\{a \rightarrow Fx, b \rightarrow Fx\} \stackrel{abs.}{\rightarrow} c \rightarrow Fx$

We can also concatenate different normative effects for the same set of situations:

$\{a \rightarrow Fx, a \rightarrow Fy\} \stackrel{abs.}{\rightarrow} a \rightarrow Fx \vee Fy$ or:

$\{a \rightarrow Fx, a \rightarrow Fy\} \stackrel{abs.}{\rightarrow} a \rightarrow Fz$

On the one extreme one can simply enumerate all desirable or undesirable generic situations[6], on the other extreme one can try to describe them at the highest abstraction level possible. For some purposes concrete rules, that specifically refer to all kinds of situations and define normative positions for them, are more suitable, for other purposes abstract rules may be preferred.

A fourth mechanism for generating paraphrases is using *deontic transformations*, e.g. $P(x) = \neg O(\neg x)$ and $F(x) = O(\neg x)$. Besides literally using the negation in some situation $(\neg x)$, the domain (world) taxonomies can be used for explicitly stating the negation of (x), e.g. in our example domain $O(educational\_use(c, st))$ can be restated as: $F(hacking(c, st) \vee commercial\_use(c, st) \vee playing(c, st))$ besides $F(\neg educational\_use(c, st))$[7].

Using these, and possibly other mechanisms it is theoretically possible to generate numerous – and if we take different natural language formulations into account, an infinite number of – paraphrases of a deep structure of a normative system.

5.1 Examples

Based upon the definitions of desirable and undesirable behaviour, we can choose from which option we want to regulate our world. In the previous paragraph we have seen that the definition of W_d is the simplest.

The formalization we can use for primary rules provides us with applicability grounds (conditions) and normative descriptions (conclusions). The conditions of the primary rules correlate to the prescriptions of the sets in W_d, whereas the conclusions of the primary rules must yield only the desired normations from W_d. Since we take the world of desirable behaviour as our starting point, the behaviour mentioned in the definition of W_d is permitted behaviour. The definitions after the |-sign provide us with the applicability grounds of the possible rules. We can chose one of the set $\{co, s, p\}$ for quantification. This choice reflects a viewpoint. The most abstract rule we can formalize stays close to the deep structure:

$\forall_c \exists_{st,co} CSL\_Computer(c) \wedge Student(st) \wedge CSL\_Course(co) \rightarrow$
$\quad Permitted(educational\_use(c, st) \wedge take\_course(co, st))$

$\forall_c \exists_p Non\_CSL\_Computer(c) \wedge People(p) \rightarrow$
$\quad Permitted(use(c, p))$

but since the quantification in the first rule is over c, take_course is actually an applica-

[6]Of course, in reality this is not feasible for most domains, since they are too large and complex.

[7]Enumeration is possible because the number of subtypes of *use* is finite.

tion ground:

$$\forall_c \exists_{st,co} CSL\_Computer(c) \wedge Student(st) \wedge CSL\_Course(co) \wedge take\_course(co,st) \rightarrow Permitted(educational\_use(c,st))$$

From our superficial relations we know that all permissions must be exceptions to a general rule. We took W_d as a starting point, so all the other situations are forbidden (W_u). Quantifying over the computers is useful when the regulation is drafted for the staff of the department. For the students, quantification over Student can be selected. The level of abstraction over facts is determined by choosing a level in the hierarchical knowledge. We can say that educational_use is permitted, but equally, we can enumerate that playing, hacking and commercial_use are forbidden:

$$\forall_p \exists_c People(p) \wedge Computer(c) \rightarrow Permitted(use(c,p))$$

$$\forall_p \exists_c People(p) \wedge CSL\_Computer(c) \rightarrow Forbidden(use(p,c))$$

$$\forall_{st} \exists_{c,so} Student(st) \wedge CSL\_Computer(c) \wedge take\_course(co,st) \wedge CSL\_Cource(co) \rightarrow Permitted(use(c,st))$$

$$\forall_{st} \exists_{c,co} Student(st) \wedge CSL\_Computer(c) \wedge take\_course(co,st) \wedge CSL\_Cource(co) \rightarrow Forbidden(playing(c,st))$$

$$\forall_{st} \exists_{c,co} Student(st) \wedge CSL\_Computer(c) \wedge take\_course(co,st) \wedge CSL\_Cource(co) \rightarrow Forbidden(hacking(c,st))$$

$$\forall_{st} \exists_{c,co} Student(st) \wedge CSL\_Computer(c) \wedge take\_course(co,st) \wedge CSL\_Cource(co) \rightarrow Forbidden(commercial\_use(c,st))$$

The first rule starts high in the taxonomy of people and use. The second rule and third rule form the permitted state (W_d), in which the exceptions are written out in the last three rules. For the abstraction of the last three rules to educational_use, the terminological knowledge concerning these actions must naturally be present explicitly. This results in the following rule:

$$\forall_{st} \exists_{c,co} Student(st) \wedge CSL\_Computer(c) \wedge CSL\_Course(co) \wedge take\_course(co,st) \rightarrow Forbidden(\neg educational_use(c,st))$$

5.2 Use of legal paraphrases

When we abstract a deep structure from an existing regulation, we are able to *study the essential normative effects* of the regulation. In the previous paragraph we have seen that there are various ways of formulating rules on the basis of the same deep structure. We can study the normative effects of existing regulations by looking from different viewpoints or abstraction levels. Looking from different viewpoints for different categories of agents, objects or actions may improve the understandability for specific people.

To *compare two laws*, we can formulate deep structures for both. When we compare the deep structures instead of the regulations themselves, part of the analysis has actually already been performed in formulating the deep structures. We can directly compare which behaviour they qualify as desirable or undesirable.

In *legislative drafting*[8] we can translate normative requirements for a world which is to be regulated into the formalization of a deep structure, where the four different dimensions lead to various alternatives for codification. Just as formalization forms the bridge step between conceptualization and design in both software engineering and knowledge

[8] See [Svensson *et al.*, 1993] for a discussion on the advantages of automated legislative drafting.

engineering, deep structures from the bridge between normative goals and codification. In system design alternative designs are possible on the basis of the formalizations, and subsequently implementation in a certain programming language follows (see for instance [vanVliet, 1984]). The deep structure can be translated into several alternative regulations (cf. system designs). As the paraphrases of regulations are still formulated in a formal representation (see paragraph 5.1), we must subsequently codify them into natural language. The deep structure ensures that all alternative codifications regulate the same world in the same way, and we can employ each of them.

6 Discussion and further research

Both in legal theory and in the field of Artificial Intelligence and Law, people have stated that legal norms should be distinguished from their expression in rules. [Brouwer, 1990] presents conceptual models of norms that intend to capture the semantics, rather than the formulation of norms. [vanKralingen *et al.*, 1993] build on Brouwer's notion and develop the notion of norm frames as an intermediate language for representing 'laws'. However, both approaches do not clearly separate the world to be regulated from the normative content of legislation. Therefore they do not allow explicit reasoning about the 'world', matching specific situations in the world to generic situations in legislation, etc. [Hage, 1993] distinguishes the natural language formulation of law (statutes) from the "mental model" that the legislator has in mind when drafting the law. The mental model consists of cases to which the legislator intends the law to apply, and those cases to which the law should not apply. These cases form the *extensional representation* of law. The natural language representations in the form of legal rules are the *intensional representation* of law. [Hage, 1993] suggests the use of a formal intensional representation to bridge the gap between the natural language intensional representation and the extensional representation of law. He does not specify what formal language to use, nor what the extensional representation looks like, but is mainly concerned with tools for for drafting new legislation that keep track of relations between different elements of the different representations of law. Our notion of 'deep structure' amounts to the extensional representation of legislation, the surface structures to different formal intensional representations.

We aim at a formalism for deep structures of law. Deep structures offer a unique description of law, they are independent of the actual layout and formulation of a regulation. When we look into the structures of law, we can do so from two different viewpoints. On the one hand, we can *study the normative effects* of a regulation by constructing an abstract description of its normative essence. The purpose of such an examination may be to compare the actual normative contents of a regulation with the original intentions of the legislative body and of the legal drafters. Here we can use abstraction in a legislative function. On the other hand, we can operate from a *compositional* viewpoint. We can study the building blocks of regulations, and the way in which they are combined to form regulations.

In legislative drafting deep structures allow the separate specification of the effects of regulations. We have postulated four dimensions along which different paraphrases can be generated. When several paraphrases are feasible for the same deep structure legal experts may select the most desirable codification. The resulting surface structures differ with respect to readability, maintenance, learnability, and the facilitation for specific purpose or target group. Should we want to automate the process we need to control the process to make it computationally feasible. We have proposed a structured drafting method for legislation. In [Ryan, 1991] the specification of norms is discussed, but there no intermediate phases are recognized. In literature on legal drafting, we foremost see requirements on the layout of law texts, as [Eijlander, 1993] poses, there is no certified method for legal drafting. We see that conceptualization roughly corresponds to research of the world to be regulated. The domain knowledge is essential for the formulation of deep structures and formal regulations. The most computerized approach we have found to legislative drafting is [Overhoff & Molenaar, 1991]. However, they use decision tables as means of formalization, which is a very static way to represent legal rules and legal reasoning. We believe that the formalization of deep structures should be close to expressions of legally qualified situations. The formalization of legal rules should allow the same structures and expressions as the natural codes (cf. [denHaan, 1994b]).

At the current stage of this research we have a good insight of the superficial structural elements of regulations and how they are combined, see for instance [denHaan, 1993] and [denHaan, 1994b]. What we aim at is a full method for the formalization and construction of deep structures, and for the generation of paraphrases. Another aspect we want to pursue is the incorporation of case law. In the collection of cases we see standard cases which only consist of filled in rules, but also cases which define (new) contexts. The additional contexts, and also cases in which rules have been applied in a new, alternative way, add to the normative effects of regulations. The situations described in cases and their respective normative qualifications must then also be added to the deep structure of the regulation. We can generate paraphrases from the new deep structure that incorporate the effects of case law.

It may be clear that there is still a lot of work to be done, but we believe that the aims and directions of our research are already very promising and interesting.

References

J.A. Breuker and N. den Haan. Separating world and regulation knowledge: where is the logic? In M. Sergot, editor, *Proceedings of the third international conference on AI and Law*, pages 41–51, New York, NJ, 1991. ACM.

P.W. Brouwer. *Coherence in Law: An Analytical Study.* PhD thesis, University of Leiden, Groningen, The Netherlands, June 1990. In dutch.

N. den Haan. Investigations into the Application of Deontic Logic. In *Workshop Executable modal and Temporal Logics at the 13th Interational Joint Conference on Artificial Intelligence (IJCAI-93)*, 1993. To be published by Springer-Verlag as a volume in the -Lecture Notes in Artificial Intelligence- series.

N. den Haan. Automated Legislative Drafting. In J.A. Breuker, editor, *Proceedings of the workshop Artificial Normative Reasoning at the European Conference on Artificial Intelligence*, pages 106–123, Amsterdam, 1994.

N. den Haan. On structure preserving representations of law. In T.F. Gordon, editor, *Proceedings of the German Conference on Artificial Intelligence and Law*, Vienna, 1994. Invited lecture.

Ph. Eijlander. *De wet stellen. Beschouwingen over onderwerpen van wetgeving.* W.E.J. Tjeenk Willink, Schoordijk Instituut - Centrum voor Wetgevingsvraagstukken, 1993.

J. Hage. An information network for legislative engineering. In *Intelligent Tools for Drafting and Computer-Supported Comparison of Law – Sixth International Conference on Legal Knowledge-based Systems Legal Knowledge-Based Systems, JURIX–1993*, pages 43–52. Koninklijke Vermande, 1993.

P. Hammond, J. Wyatt, and A. Harris. Drafting protocols, certifying clinical trial designs and monitoring compliance. In J.A. Breuker, editor, *Proceedings of the workshop Artificial Normative Reasoning at the European Conference on Artificial Intelligence*, pages 124–131, Amsterdam, 1994.

H.L.A. Hart. *The Concept of Law.* Clarendon Press, Oxford, 1961.

Mr. R.W. Overhoff and Mr. L.J. Molenaar. *In de regel beslist. Een beschouwing over regelgeving met behulp van beslissingstabellen.* SDU Uitgeverij, Plantijnstraat, 's-Gravenhage, 1991.

J.A.F. Peters and F.M.C. Vlemminx. Wet, symbool en teken. In W.J. Witteveen, P. Van Seters, and G. van Roermund, editors, *Wat maakt de wet symbolisch?*, pages 37–53. W.E.J. Tjeenk Willink, Katholieke Universiteit Brabant, Faculteit der Rechtsgeleerdheid, 1991.

M. Ryan. Towards specifying norms. In J.-J. Ch. Meyer and R.J. Wieringa, editors, *Proceedings of the First International Workshop on Deontic Logic in Computer Science, DEON '91*, pages 335–346, Amsterdam, 1991.

G. Sartor. Normative Conflicts in Legal Reasoning. *Artificial Intelligence and Law*, 1(2-3):209–235, 1992.

A. Soeteman. *Norm en Logica.* Tjeenk Willink, Zwolle, The Netherlands, 1981. ISBN 90-271-1801-9.

J.S. Svensson, J.G.J. Wassink, and B. van Buggenhout, editors, *Legal Knowledge-based Systems: Intelligent Tools for Drafting and Computer-Supported Comparison of Law*, 1993.

R. van Kralingen, E. Oskamp, and E. Reurings. Norm frames in the representation of laws. In *Intelligent Tools for Drafting and Computer-Supported Comparison of Law – Sixth International Conference on Legal Knowledge-based Systems Legal Knowledge-Based Systems, JURIX–1993*, pages 11–22. Koninklijke Vermande, 1993.

J.C. van Vliet. *Software Engineering.* Stenfert Kroese B.V., Leiden/Antwerpen, 1984.

G.H. von Wright. On the logic of norms and actions. In Risto Hilpinen, editor, *New Studies in Deontic Logic*, pages 3–35. D. Reidel, Dordrecht, 1981.

Mr. C. Waaldijk. *Wetgevingswijzer.* Koninklijke Vermande B.V. -uitgevers-, Lelystad, 1985.

J. Zeleznikow and D. Hunter. An overview of some reasoning formalisms as applied to law. *Think*, 3:24–40, June 1994. Special issue on Artificial Intelligence and Law.

Acknowledgements

The ideas about deep structures in law have materialized in discussions with Joost Breuker of the Department of Computer Science and Law. The authors would also like to thank the anonymous referees, Henry Prakken and Wiepke Westbroek for their valuable comments on earlier drafts of this paper.